UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, STATION A, MONTRÉAL, QUÉBEC, H3C 3J7

LES MODELES PROBABILISTES APPLIQUES AUX PROBLEMES DE TRANSPORT ET D'ORDONNANCEMENT

par

MICHEL TRAHAN

Cette recherche appliquée a été subventionnée par "Ford Motor Company of Canada".

Université de Montréal
Publication # 185 - Département d'informatique
Publication # 12 - Centre de Recherche sur les Transports
Mai 1974

TABLE DES MATIERES

Chapitre		page
	SOMMAIRE	vi
I	INTRODUCTION	1
1.1	Motivations	1
1.2	Problèmes de transport et modèles probabilistes	2
1.3	Notre contribution aux problèmes de transport	9
1.4	Problèmes d'ordonnancement et modèles probabilistes	10
1.5	Notre contribution aux problèmes d'ordonnancement	13
II	UN ALGORITHME D'AFFECTATION PROBABILISTE	15
2.1	Notation et définitions	15
2.2	Algorithme	20
2.3	Justification de l'algorithme	24
2.4	Exemple et résultats numériques	26
2.5	Performance de l'algorithme	31
III	UN MODELE D'AFFECTATION DYNAMIQUE	34
3.1	Historique et commentaires	34
3.2	Avertissement	36
3.3	Notation, définitions et exigences	37
3.4	Modèle d'affectation dynamique	38
3.5	Evaluation des fonctions de densité	41
3.6	Partition des temps de départ	46
3.7	Etude de la moyenne et de la variance de f_v	49

Chapitre			page
	3.8	Applications	51
	3.9	Conclusion	54
IV		L'EVALUATION DE LA MATRICE ORIGINE-DESTINATION ET DE CERTAINS ELEMENTS DU RESEAU	55
	4.1	Historique	55
	4.2	Affectation des voyages	55
	4.3	Mesure de concordance	58
	4.4	Modèles d'affectation proportionnels	60
	4.5	Modèle de gravité	62
	4.6	Modèle de régression linéaire multiple	65
	4.7	Modèles d'affectation non proportionnels	66
	4.8	Généralisation et conclusion	67
V		L'ESTIMATION DE LA DISTRIBUTION DE LA DUREE D'UN PROGRAMME	69
	5.1	Représentation d'un programme par un graphe	69
	5.2	Approche probabiliste pour l'étude des réseaux PERT	70
	5.3	Sommaire de deux approches existantes pour estimer la distribution de la durée d'un programme	73
	5.4	Borne inférieure pour la fonction de distribution de la durée d'un programme	75
	5.5	Exemples et commentaires	80
	5.6	Réseaux PERT "décisionnels"	86
	5.7	Probabilité qu'un sommet soit atteint	92
	5.8	Borne inférieure pour la fonction de distribution de la durée d'un programme représenté par un RPD	93

Chapitre		page
VI	DES BORNES INFERIEURES POUR LA DUREE MOYENNE D'UN PROGRAMME	97
6.1	Notation et définitions	97
6.2	Enumération de certaines bornes inférieures pour la durée moyenne d'un programme	98
6.3	Elaboration de nouvelles bornes	102
	CONCLUSION	109
	APPENDICE I	112
	APPENDICE II	113
	APPENDICE III	115
	APPENDICE IV	158
	APPENDICE V	162
	REMERCIEMENTS	167
	BIBLIOGRAPHIE	168

SOMMAIRE

Les modèles probabilistes suscitent un grand intérêt en recherche opérationnelle: ils permettent d'étudier des phénomènes qui n'étaient pas contrôlables de façon satisfaisante suivant un point de vue déterministe. Ces modèles, soutenus par la théorie des réseaux, laissent entrevoir la possibilité de mieux contrôler les problèmes de transport et d'ordonnancement. La présente thèse a pour but de collaborer au développement des modèles probabilistes à l'intérieur de ces deux problèmes.

Relativement aux problèmes de transport, nous élaborons un algorithme fondé sur un modèle d'affectation probabiliste sans capacité; ce même modèle sert à développer un modèle d'affectation probabiliste dynamique permettant une étude dans le temps du comportement des voyageurs. En outre, nous montrons qu'il est possible de substituer à l'enquête origine-destination utilisée lors de la distribution des voyages, des données de trafic pouvant être obtenues simplement. Cette approche permet de résoudre d'un coup les problèmes de génération, de distribution et d'affectation des voyages.

Relativement aux problèmes d'ordonnancement, cette thèse apporte des améliorations techniques à la méthodologie PERT: nous présentons une façon d'estimer la fonction de distribution de la durée d'un programme ainsi que ses différents moments. La théorie s'applique par extension aux réseaux PERT "décisionnels". Enfin, nous proposons une approche pour obtenir des bornes inférieures à la durée moyenne d'un programme.

CHAPITRE I

INTRODUCTION

1.1 Motivations

La construction d'un modèle a pour but de schématiser logiquement les complexités et les incertitudes d'un problème afin de permettre une analyse plus compréhensible et plus simple. Les modèles que nous retrouvons en recherche opérationnelle sont probabilistes ou déterministes selon que le modèle contient ou ne contient pas de composantes aléatoires. L'introduction de composantes aléatoires peut se justifier par les spécifications fondamentales du modèle ou pour tenir compte des écarts entre les prédictions du modèle et l'observation des données. En recherche opérationnelle, nous sommes le plus souvent confrontés avec des problèmes fondés sur des phénomènes comportant des incertitudes, qu'il s'agisse de transport, de génie, d'économie ou de toute autre activité. Ces incertitudes peuvent être, entre autres, l'instabilité de l'environnement, les bruits dans le système ou encore le comportement inattendu ou inexpliqué de l'homme. Les modèles probabilistes tentent de mesurer à l'aide de probabilités ces incertitudes lorsque le phénomène laisse entrevoir un degré de régularité. C'est pour ces raisons que les modèles probabilistes ont suscité un grand intérêt en recherche opérationnelle, et particulièrement en ce qui a trait aux problèmes pouvant être représentés par un réseau.

Nous nous proposons dans cette thèse d'étudier deux problèmes précis faisant appel aux modèles probabilistes dans la théorie des réseaux,

soit celui de l'affectation du trafic dans le transport routier ou celui de l'évaluation du temps total requis pour réaliser un projet.

1.2 Problèmes de transport et modèles probabilistes

Aujourd'hui, un des grands défis qui se pose à l'homme est de développer des villes qui offrent à la collectivité un environnement sain. La planification en transport constitue un élément clé de la réponse à ce défi. Les planificateurs en transport ont pour objectif, entre autres, de développer des modèles de prédiction du trafic en vue d'orienter et de justifier un plan de développement de transport. L'élaboration d'un tel modèle s'effectue suivant un ensemble d'opérations pouvant être énumérées chronologiquement ainsi:

a) la génération des voyages: l'expression "génération des voyages" est utilisée pour décrire le nombre de voyages débutant et se terminant dans une zone en rapport avec l'utilisation des espaces et les facteurs socio-économiques inhérents à la zone;

b) la distribution des voyages: la deuxième étape dans l'analyse du trafic traite du processus qui estime où les voyages débutant dans une zone se terminent;

c) le choix d'un mode de transport: connaissant la distribution des voyages, il faut conséquemment développer un modèle représentant la répartition des voyages entre les différents modes de transport disponibles;

d) l'affectation des voyages: l'affectation des voyages est la détermination des routes empruntées par les usagers d'un certain mode de

transport se déplaçant d'une zone à une autre et du nombre d'usagers sur chacun des tronçons de routes qui en résulte;

e) l'analyse des résultats: à cette étape, on évalue globalement la qualité du modèle élaboré en comparant les prédictions faites par le modèle et les observations tirées de la réalité.

L'affectation des voyages, admettant l'hypothèse que la distribution des voyages est connue, revient à un problème de flot à plusieurs commodités; il s'agit donc là d'un problème dépendant de la théorie des réseaux et nous allons nous intéresser particulièrement à cette étape de l'analyse en transport au cours des chapitres II et III.

A cette phase de la planification en transport, nous dérisons un modèle reproduisant le comportement des usagers devant le choix d'une route les menant d'une origine à une destination. Le nombre d'usagers empruntant une route spécifique entre deux zones dépend de la structure du réseau, des coûts d'utilisation associés aux tronçons de routes, de la perception de ces coûts par les usagers, du nombre total d'usagers présents sur les éléments du réseau et enfin du choix même des routes. Il est impossible de simuler exactement le processus qui régit le comportement des usagers face à ce choix. Néanmoins, il est réaliste de supposer que les usagers tentent de minimiser leur coût de parcours. Cette hypothèse implique que la répartition finale du trafic doit répondre au principe suivant: les coûts sur les différentes routes entre une origine et une destination sont approximativement égaux pour les routes utilisées et moindres que sur les routes non utilisées. Ce principe a été énoncé par Wardrop (1952). L'éventail des modèles d'affectation que permet le terme "approximativement" apparaissant dans le principe de Wardrop est étendu;

nous allons le parcourir rapidement.

Un premier tri* dans les types de modèles d'affectation s'effectue naturellement selon que l'on exige que les coûts soient exactement les mêmes sur les routes utilisées ou selon que l'on interprète le terme "approximativement" avec beaucoup de latitude. Le premier groupe réunit les modèles d'équilibre et le second, les modèles de dispersion. Les modèles d'équilibre reposent sur les hypothèses suivantes, à savoir que les coûts sur les tronçons de routes croissent lorsque la charge totale sur ces tronçons de routes augmente, que le trafic est composé d'une seule catégorie d'usagers percevant de façon identique les coûts d'utilisation et que les usagers sur un tronçon de route se partagent également son coût d'utilisation. Ainsi, les modèles d'équilibre ne considèrent pas les usagers individuellement avec leurs caractéristiques propres. Les modèles de dispersion impliquent les mêmes hypothèses, sauf qu'ils présument que les usagers perçoivent de façon différente les coûts des tronçons de routes. Ces modèles veulent illustrer le comportement non toujours optimal des usagers en termes des coûts assignés aux éléments du réseau.

L'affectation du trafic a une influence réelle sur les coûts associés aux éléments du réseau. En effet, quand le trafic est dense sur un tronçon de route, il se crée de la congestion et il en résulte que le coût du voyage pour un usager empruntant ce tronçon de route est plus élevé que s'il était seul sur le réseau. Il est connu que le retard causé par la congestion a pour effet de faire dévier certains usagers vers des routes qui deviennent alors optimales; dans certains cas, des

* voir Appendice I

usagers annuleront même leur voyage. Les modèles d'affectation qui tiennent compte de la congestion sont préférables, mais aussi plus dispendieux et plus compliqués.

On reconnaît une autre division dans les deux groupes de modèles précédents: les modèles avec capacités faisant l'hypothèse supplémentaire qu'il existe une relation entre la charge sur un tronçon de route et son coût ainsi que les modèles sans capacité qui négligent l'effet de la charge d'un tronçon de route sur son coût.

A l'intérieur des modèles d'équilibre avec capacités, nous observons deux tendances: les modèles heuristiques et les modèles mathématiques. Les modèles heuristiques sont moins strictes que les modèles mathématiques. Par un processus itératif fondé sur des éléments plutôt intuitifs, ces modèles tentent de se rapprocher de l'état d'équilibre. A chaque itération, un algorithme du plus court chemin est utilisé. Le processus se termine après un nombre arbitraire d'itérations. Ces modèles utilisent des concepts très simples et dans bien des cas, il est possible de suivre chacune des étapes de l'affectation du trafic. L'inconvénient majeur réside dans le fait qu'il est impossible d'évaluer l'écart entre la répartition du trafic à la fin du processus itératif et celle à l'état d'équilibre. Les premiers à étudier ces modèles furent Mosher (1963), Schneider (1963), le Bureau of Public Roads (1964), Martin (1964), Martin et Manheim (1965) et Overgaard (1967). En général, ces modèles donnent une solution non unique et variant selon l'algorithme utilisé; ilsrequièrent peu de temps machine, mais ne permettent pas d'évaluer rapidement les effets d'une légère modification des données initiales du problème sur la répartition de trafic obtenue à la fin du processus itératif.

Les modèles mathématiques formulent le problème de l'affectation du trafic comme un problème de programmation mathématique. La solution réalisable optimale au problème de programmation associé correspond à l'affectation recherchée du trafic. La fonction économique associée au modèle mathématique énonce implicitement qu'à l'optimum, aucun usager ne peut diminuer son coût de parcours en changeant de route. La programmation mathématique minimise la fonction économique soumise aux contraintes de la conservation du flot. L'équivalence entre l'affectation du trafic à l'état d'équilibre sur le réseau routier et la minimisation d'une certaine fonction connexe a été originellement démontrée par Beckmann *et al* (1956). Contrairement aux modèles heuristiques, ces modèles rendent généralement une solution unique, nécessitent beaucoup d'espace mémoire et de temps machine; ils permettent en outre la postoptimisation à coût raisonnable.

Récemment, des algorithmes itératifs basés sur des critères d'améliorations rationnels ont été développés pour obtenir la répartition du trafic à l'état d'équilibre. Ces algorithmes furent proposés, entre autres, par Gibert (1968), Murchland (1969), Dafermos et Sparrow (1969), Wilkie et Stephanek (1971), Leventhal *et al* (1973) et par Nguyen (1973). Nguyen (1973) a développé différents algorithmes pour les modèles d'équilibre avec capacités et présenté des résultats comparatifs pour divers algorithmes effectuant l'affectation du trafic sous les modèles d'équilibre avec capacités.

En ce qui a trait aux modèles d'équilibre sans capacité, nous retrouvons aussi deux tendances. La première est basée sur la programmation linéaire, ce qui est un cas particulier des modèles mathématiques. La seconde tendance opte pour le modèle du "tout ou rien" qui fait passer tous les usagers voyageant entre une origine et une destination par la

route à coût minimal. Ces modèles sont très répandus à cause de leur simplicité et de leur faible coût d'implantation. Il n'en reste pas moins que ces modèles donnent une approximation très imprécise de la réalité, sauf lorsqu'il n'y a pas congestion, ce qui est le plus souvent le cas en milieu rural. Tillman *et al* (1968), le Urban Transportation Planning (1972) et Wildermuth (1972) ont d'ailleurs étudié ces modèles. Cette brève rétrospective des modèles d'équilibre nous amène à considérer de même les modèles de dispersion.

La totalité des modèles de dispersion sans capacité répartit le trafic entre les routes "efficaces" sans considération de la charge sur les tronçons de routes. Ils se divisent en trois groupes: les modèles basés sur les courbes de dispersion, les modèles énumérant toutes les routes et les modèles probabilistes. Les modèles basés sur les courbes de dispersion furent populaires au début des années 40 avant l'utilisation à une grande échelle des ordinateurs. Pour prévoir le taux d'utilisation des routes, les planificateurs construisaient des courbes de dispersion à partir de considérations théoriques mettant en évidence le coût relatif des différentes routes. Les principaux investigateurs de ces modèles furent Brown (1948), Campbell (1951), Trueblood (1952) et plus tard Moskowitz (1956). Les modèles utilisant l'énumération, comme développés par Irwin *et al* (1961), McLaughlin (1966) et Sakarovitch (1968), identifient toutes les routes se situant entre une origine et une destination ainsi que leur coût respectif. Les usagers sont alors répartis entre les routes de la façon la plus probable, suivant la relativité des coûts sur ces routes. Enfin, les modèles probabilistes répartissent les usagers entre les routes possibles à partir de probabilités reflétant la vraisemblance qu'un usager

utilise une route plutôt qu'une autre. Les principaux avantages de ces modèles sont d'éviter l'énumération des routes et de répartir simultanément tous les usagers sur les routes dites "efficaces". Stopher (1969), Dial (1971), Schneider (1973) et Robillard (1973) ont travaillé sur ces modèles. Il est à noter que les chapitres II et III portent sur ces modèles.

Les modèles de dispersion avec capacités comportent des hypothèses de départ très attrayantes; malheureusement, très peu d'algorithmes s'appuyant sur ces modèles ont été proposés. Les seules tentatives, et timides encore, résident dans l'amélioration, à l'aide de processus itératifs, des approches que l'on retrouve dans les modèles de dispersion sans capacité. Par exemple, une approche améliorée affecte successivement des pourcentages de la demande en voyages entre chaque paire origine-destination suivant un modèle de dispersion sans capacité, où après chaque affectation partielle on réajuste les coûts. Wildermuth (1972) et Schneider (1973) se sont penchés sur ces modèles.

Enfin, il y a un groupe particulier de modèles se situant entre les modèles d'équilibre et les modèles de dispersion. Ces modèles couvrent les cas où on impose une borne supérieure pour la charge sur les tronçons de routes. Une des approches utilise la programmation linéaire avec des contraintes supplémentaires associées aux bornes, comme le proposent Charnes et Cooper (1961) ainsi que Tomlin (1966). Une autre approche, fondée sur le "tout ou rien" et développée par Ford et Fulkerson (1957) ainsi que par Homburger (1969), affecte le plus d'usagers possible sur la route la moins dispendieuse tant qu'il n'y a pas saturation sur un tronçon de route. Lorsque la charge sur un tronçon de route atteint

sa capacité, son coût devient alors infini. On trouve alors une route à valeur minimale et on la sature à son tour. Le processus se poursuit jusqu'à ce que tous les usagers aient été affectés.

Tous les modèles présentés se limitent aux cas statiques, c'est-à-dire qu'ils simulent l'affectation du trafic en régime premanent. Les modèles dynamiques qui font l'affectation des usagers selon une répartition temporelle sont rares. Yagar (1971), Florian (1973) et Robillard (1973) ont été parmi les premiers à effectuer des recherches dans ce domaine.

Cet aperçu, pourtant non exhaustif, nous a permis de voir les différentes étapes que comporte l'analyse des transports et les différents modèles d'affectation. Nous sommes maintenant en mesure de situer la matière qui fera l'objet des chapitres II, III et IV et d'entrevoir l'originalité que nous avons tenté d'apporter à ce domaine de la recherche.

1.3 Notre contribution aux problèmes de transport

Le chapitre II présente un algorithme sous-jacent au modèle d'affectation probabiliste sans capacité suggéré par Dial (1971). L'algorithme proposé est basé sur des observations faites par Robillard (1973) concernant ce modèle et il améliore l'algorithme de Dial en réduisant le nombre d'accès aux éléments du réseau et en simplifiant les formules à évaluer.

Le chapitre III généralise le modèle d'affectation probabiliste sans capacité développé par Dial dans le cas où la demande en voyages varie avec le temps, où les temps de parcours et les vitesses sur les tronçons de routes sont des variables aléatoires et où les éléments qui com-

posent le critère de dispersion varient dans le temps. Les hypothèses introduites permettent une étude dynamique de la répartition du trafic, la prédiction des heures de pointe à différents endroits du réseau ainsi que la mesure de leur durée. Comme l'ont noté Huber, Boutwell et Withford (1968), la généralisation des modèles classiques d'affectation aux cas dynamiques est une étape qu'il faut franchir pour acquérir une meilleure compréhension du phénomène du transport.

Le chapitre IV se penche sur le problème de la distribution des voyages. De façon générale, la distribution des voyages s'effectue à partir d'une enquête origine-destination; cette dernière est coûteuse, demande beaucoup de temps et conduit souvent à des résultats imprécis. Nous démontrons qu'il est possible de substituer à cette enquête l'utilisation de données de trafic qui peuvent être obtenues simplement et qui sont disponibles dans la plupart des grandes villes. Une analyse bien menée de ces données permet de résoudre d'un coup les problèmes de génération, de distribution et d'affectation des voyages. Une attention particulière est accordée aux modèles d'affectation de type probabiliste.

1.4 Problèmes d'ordonnancement de modèles probabilistes

Parmi les méthodes élaborées ces dernières années dans le but de mettre en oeuvre et de surveiller l'exécution de programmes de production ou de recherches, les méthodes CPM (Critical Path Method) et PERT (Program Evaluation and Review Technique) tiennent une place dominante par leur simplicité, leur efficacité et la richesse des développements qu'elles ne cessent d'apporter. La base mathématique de ces deux méthodes est consti-

tuée par la théorie des graphes et celle des réseaux. Ces méthodes, fondées sur des modèles faisant appel aux réseaux, ont connu un essor important et ont suscité un grand intérêt. Il n'en reste pas moins que certains problèmes d'ordonnancement posent encore des difficultés lorsque les durées des activités composant le programme sont des variables aléatoires. Aussi, allons-nous nous restreindre aux modèles probabilistes d'ordonnancement et tout particulièrement à l'étude de la fonction de distribution associée à la durée totale d'un programme ainsi qu'aux différents moments de cette fonction de distribution.

Charnes, Cooper et Thompson (1963) furent parmi les premiers à étudier les propriétés stochastiques des réseaux PERT et ceci, en appliquant des méthodes de programmation mathématique. Obtenant une approximation pour la fonction de densité associée à la durée d'un programme pour des réseaux PERT simples, la méthode qu'ils proposent devient pratiquement inutilisable dans le cas de grands réseaux PERT où les fonctions de densité associées aux durées des activités sont quelconques. A la même époque, MacCrimmon et Ryavec (1964) ont présenté des résultats mathématiques obtenus à partir d'une analyse détaillée des hypothèses faites lors des calculs propres à la méthode PERT et des approximations qui en résultent. Martin (1965) a présenté une méthode pour évaluer la fonction de densité de la durée d'un programme lorsque les fonctions de densité associées aux durées des activités peuvent être exprimées sous forme polynômiale. Plus récemment, Hartley et Wortham (1966) ainsi que Ringer (1969) ont développé des méthodes analytiques qui estiment cette fonction de densité pour certains types de réseaux. Des techniques de simulation permettent d'appliquer cette méthode à tous les réseaux PERT. L'article de Burt et

Garman (1971) traite de la méthode de Monte Carlo appliquée aux problèmes des réseaux PERT probabilistes et introduit de nouvelles techniques analytiques pour améliorer les méthodes de simulation dans ce contexte précis. Enfin, l'article de Kleindorfer (1971) présente une méthode pour obtenir une borne inférieure et une borne supérieure pour la fonction de distribution de la durée d'un programme. Cette dernière méthode se limite en pratique aux cas où les fonctions de densité associées aux durées des activités sont discrètes.

Les méthodes disponibles pour approcher la fonction de densité de la durée d'un programme ont un champ d'application limité: il faut se restreindre à des réseaux PERT d'un certain type ou encore consentir à beaucoup de travail.

Un autre point intéressant et caractéristique des problèmes d'ordonnancement est l'évaluation du temps moyen requis pour réaliser un programme. Ce temps moyen peut être obtenu à l'aide de la fonction de densité de la durée du programme et de la définition de l'espérance mathématique. Malheureusement, comme nous l'avons déjà mentionné, l'évaluation de la fonction de densité pose certains problèmes. De ce fait, la recherche s'est orientée de façon à tenter d'obtenir une approximation pour cette moyenne, d'autant plus que, dans la majorité des cas, une valeur approximative de la moyenne est satisfaisante. Une telle approximation est obtenue par les méthodes de simulation comme l'ont affirmé Pritsker et Kiviat (1969), Moder et Phillips (1970) ainsi que Burt et Garman (1971). D'autre part, Lindsey (1971) a développé une méthode pour estimer cette moyenne en transformant le réseau PERT initial. Une borne inférieure à la moyenne est aussi disponible; en pratique, cette borne

est obtenue en remplaçant chacune des variables aléatoires par leur moyenne et en utilisant la méthode CPM. Kaufman et Desbazeille (1969) discutent de cette méthode. L'inconvénient majeur de cette méthode réside dans le fait que la borne est indépendante de la variance de la variable aléatoire associée aux durées des activités. Comme l'ont noté MacCrimmon et Ryavec (1964), cette borne peut être très mauvaise. Toutefois, la borne peut être améliorée si on utilise des techniques de simulation. Moder et Phillips (1970) résument cette approche. Quant à Fulkerson (1962), il propose une méthode pour calculer une borne inférieure à la moyenne lorsque les variables aléatoires associées aux durées des activités sont discrètes. La borne de Fulkerson est toujours meilleure que celle du CPM. Clingen (1964) généralise la méthode de Fulkerson aux cas continus et Elmaghraby (1967) améliore la borne de Fulkerson en tirant profit des caractéristiques propres aux réseaux PERT. Kleindorfer (1971) obtient une borne inférieure et une borne supérieure pour le temps moyen d'un programme dans le cas discret. Cette référence complète la liste des principaux travaux se rapportant au sujet que nous allons traiter.

1.5 Notre contribution aux problèmes d'ordonnancement

Le chapitre V propose une nouvelle façon d'étudier les réseaux PERT probabilistes et suggère une borne inférieure à la fonction de densité associée à la durée d'un programme. Nous supposons que les durées des activités sont des variables aléatoires discrètes ou continues, indépendantes entre elles et définies sur un ensemble positif borné. Cette nouvelle approche s'avère très efficace du point de vue opérationnel pour

de grands réseaux PERT dont les fonctions de densité, associées aux durées des activités, sont quelconques. En l'utilisant, on obtient directement des bornes pour les différents moments de la fonction de densité associée à la durée d'un programme. Les résultats obtenus sont appliqués aux réseaux PERT probabilistes "décisionnels" négligés jusqu'à ce jour.

Au chapitre VI, nous présentons une approche généralisée afin d'obtenir des bornes inférieures pour la durée moyenne d'un programme. Nous montrons que la borne CPM et celle proposée par Fulkerson, entre autres, tiennent lieu de cas particuliers dans cette approche fondée sur l'inégalité de Jensen. Cette approche permet d'améliorer les bornes existantes: un cas particulier est considéré où la borne proposée est toujours aussi bonne que celles de Fulkerson et d'Elmaghraby si on consent à payer un peu plus. Nous supposons que l'ensemble des temps requis pour accomplir les activités se terminant à un événement est indépendant de l'ensemble des temps associés aux activités se terminant à un autre événement. De plus, nous supposons que les temps requis pour accomplir les activités sont des variables aléatoires discrètes ou continues définies sur un ensemble positif borné.

CHAPITRE II

UN ALGORITHME D'AFFECTATION PROBABILISTE

2.1 Notation et définitions

L'objectif de ce chapitre est d'améliorer l'algorithme développé par Dial (1971) pour le modèle d'affectation probabiliste sur plusieurs chemins. Nous supposons que le lecteur est familier avec la théorie de Dial et nous utilisons sa notation.

Un réseau de transport est défini par un quadruplet [V, E, n, d] où

V est un ensemble fini, non vide de sommets;

E est un ensemble fini, non vide d'arcs orientés;

n est une fonction définie sur l'ensemble E qui fait correspondre à chaque arc e dans E une paire ordonnée de sommets distincts dans V , soit (v_i,v_j). v_i et v_j sont respectivement le sommet initial et le sommet terminal de l'arc e . n induit de façon naturelle les fonctions n_1 et n_2 définies comme suit:

$$n(e) = (v_i,v_j) = [n_1(e), n_2(e)] \quad \text{pour} \quad e \quad \text{dans} \quad E .$$

Un arc est indifféremment référé par la lettre e ou par la paire ordonnée de sommets qui lui est associée

et

d est une fonction qui affecte à chaque arc e dans E un

scalaire non négatif $d(e)$ (≥ 0). Il est entendu que $d(e)$ égale zéro pour e non dans E. La fonction d, dans ce contexte, est une mesure d'utilité associée aux arcs. d est décroissante suivant le coût de l'arc et croissante suivant son niveau de service.

De plus, nous définissons une fonction L qui affecte à chaque arc e dans E un scalaire positif (> 0) associé à la longueur (ou au coût) de l'arc.

Pour simplifier la présentation, nous supposons que le réseau possède une seule origine v_o et une seule destination v_d. Cette restriction pourra être abandonnée par la suite sans complication. Les notions de chemin et de circuit correspondent aux définitions habituelles. Ainsi, un chemin p entre v_i et v_j est une suite ordonnée de sommets $v_i = v_1, v_2, \ldots, v_\ell = v_j$ telle que (v_k, v_{k+1}) appartient à E pour $k = 1, \ldots, \ell - 1$. Nous notons par $P_{v_i v_j}$ l'ensemble des chemins entre les sommets v_i et v_j. A l'avenir, nous noterons:

$\sum_{a \text{ dans } A}$ par $\sum_{A}$;

$\prod_{a \text{ dans } A}$ par $\prod_{A}$;

$P_{v_o v}$ par P_v

et

$P_{v_o v_d}$ par P .

Nous considérons la paire origine-destination (v_o, v_d) et, pour chaque sommet v dans V, nous dénotons par $p(v)$ la longueur du plus court chemin entre v_o et v. Le modèle d'affectation peut être décrit en spécifiant pour chaque chemin p de P une quantité $Pr[p]$ représentant la probabilité que le chemin p soit utilisé pour un voyage de v_o à v_d avec

$$\sum_P Pr[p] = 1 \quad .$$

$Pr[e]$, la probabilité que l'arc e soit utilisé pour un voyage entre les sommets v_o et v_d, est égale à la somme des probabilités des chemins passant par e, c'est-à-dire:

$$Pr[e] = \sum_P \delta_{ep} \, Pr[p]$$

où

$$\delta_{ep} = \begin{cases} 1 & \text{si le chemin } p \text{ comprend l'arc } e \\ 0 & \text{autrement .} \end{cases}$$

Pour développer un algorithme d'affectation, nous devons définir les probabilités associées aux chemins. Cette définition doit permettre d'évaluer efficacement les quantités $Pr[e]$ sans impliquer pour autant l'énumération de tous les chemins. Utilisant la terminologie de Dial, nous dirons qu'un voyage débutant au sommet v_o et se terminant au sommet v_d emprunte le chemin p avec la probabilité $Pr[p]$ définie ainsi:

$$Pr[p] = K \text{ EXP } \theta[L(p) - p(v_d)] \quad \text{avec } p \text{ dans } P \quad (2.1)$$

où $L(p)$ est la longueur du chemin p

$$L(p) = \sum_E \delta_{ep} L(e) \ ;$$

θ est un paramètre de calibration

et K est une constante de normalisation.

La probabilité définie à l'équation (2.1) tient lieu de cas particulier pour une équation générale à savoir:

$$Pr[p] = K \prod_E d(e)^{\delta_{ep}} \ . \quad (2.2)$$

Cette nouvelle définition de la probabilité associée aux chemins généralise celle proposée par Dial et permet, comme nous le verrons, d'effectuer l'affectation de façon efficace. Pour retrouver la formulation de Dial, il suffit de définir $d(e)$ de la façon suivante:

$$d(e) = \text{EXP } \theta L(e) \ .$$

Dans la plupart des cas, nous ne sommes pas intéressés à identifier individuellement chacun des voyages, mais nous désirons connaître le flot total sur chacun des arcs du réseau. Si $Y_{v_o v_d}$ est le nombre de voyages entre v_o et v_d, le nombre de voyages utilisant l'arc e est donné par $Y_{v_o v_d} * Pr[e]$; c'est pourquoi nous allons nous attarder à l'évaluation de $Pr[e]$. En se servant de la définition générale de $Pr[p]$, Robillard (1973) a démontré qu'il était possible d'écrire l'égalité suivante:

$$Pr[e] = \frac{w[v_o,n_1(e)]\, d(e)\, w[n_2(e),v_d]}{w(v_o,v_d)} \qquad (2.3)$$

où

$$w(v_i,v_j) = \begin{cases} 1 & \text{si } v_i = v_j \\ 0 & \text{si } P_{v_i v_j} \text{ est vide} \\ \sum_{P_{v_i v_j}} \prod_{e' \text{ dans } E} d(e')^{\delta_{e'p}} & \text{autrement.} \end{cases}$$

L'avantage d'exprimer $Pr[e]$ en fonction des w vient du fait que ces fonctions peuvent être évaluées de façon récursive:

$$w(v_i,v_j) = \sum_V w(v_i,v)\, d(v,v_j) \quad \text{pour tous les sommets } v_i,v_j \text{ dans } V \qquad (2.4)$$

et

$$w(v_i,v_j) = \sum_V w(v,v_j)\, d(v_i,v) \quad \text{pour tous les sommets } v_i,v_j \text{ dans } V\,. \qquad (2.5)$$

Les équations (2.4) et (2.5) permettent d'évaluer les $w[v_o,n_1(e)]$ en parcourant dans un ordre particulier les sommets et de calculer les $Pr[e]$ en parcourant dans l'ordre inverse ces mêmes sommets.

Ces probabilités sont bien définies en autant que le réseau sur lequel on affecte les voyages ne contient pas de circuits orientés. Dial traite cette contrainte en introduisant la notion de chemin "efficace". En fait, nous pouvons définir un chemin "efficace" comme bon nous semble,

en autant que la définition adoptée permette d'éliminer temporairement certains arcs du réseau de sorte que le réseau résultant ne contienne pas de circuits orientés. Selon une des définitions de chemin "efficace" proposées par Dial, un arc e est éliminé du réseau initial lorsque $p[n_1(e)]$ est plus grand ou égal (>) à $p[n_2(e)]$. Cette élimination s'effectue seulement pour l'origine v_o . Ainsi, l'affectation se fait toujours sur un réseau sans circuit orienté relativement à une paire origine-destination.

2.2 Algorithme

Nous considérons maintenant le problème de l'affectation des voyages entre une origine v_o et plusieurs destinations v_i . Nous convenons de dénoter par $Y(v_i)$ le nombre de voyages débutant au sommet v_o et se terminant au sommet v_i . NBDEST représente le nombre de destinations, c'est-à-dire le nombre de i pour lesquels $Y(v_i)$ est positif. Pour avoir un réseau sans circuit orienté, nous éliminons implicitement les arcs pour lesquels $p[n_1(e)] \geq p[n_2(e)]$.

Afin de permettre une présentation claire, nous avons divisé l'algorithme en deux étapes. Dans la première étape, appelée ***BALAYAGE AVANT***, les quantités $p(v)$, $d(e)$ et $w(v_o,v)$ sont évaluées alors que les sommets du réseau sont parcourus dans un ordre similaire à celui utilisé dans l'algorithme de Dijkstra, présenté par Hu (1969), pour calculer la longueur du plus court chemin entre deux sommets d'un réseau. Dans la seconde étape, appelée ***BALAYAGE ARRIERE***, les sommets sont parcourus dans l'ordre inverse de la première étape et les voyages sont distribués sur

les arcs rencontrés. Exposons maintenant les détails de l'algorithme.

a) *BALAYAGE AVANT*

Lors du *BALAYAGE AVANT*, nous calculons les valeurs d(e) et $w[v_o, n_1(e)]$ pour les arcs pertinents de E et nous construisons itérativement un ensemble ordonné de sommets dénoté par X qui contiendra éventuellement toutes les destinations. $\bar{X}$ est le complément de X : $V = X \cup \bar{X}$. Au départ, v_o est le seul sommet dans X et les variables Y(v) et NBDEST sont fixées.

INITIALISATION

Nous posons :

p(v):= M pour tous les sommets v dans V

et

F(e):= 0 pour tous les arcs e dans E .

La variable p(v) contiendra la longueur du plus court chemin entre v_o et v , et la variable F(e), le nombre de voyages empruntant l'arc e . M est une constante contenant le plus grand nombre pouvant être mémorisé par l'ordinateur utilisé.

Nous initialisons les variables suivantes:

NODE := v_o,

SUMDEST := 0

et p(NODE) := 0 .

EVALUATION

Pour tous les sommets v dans $\bar{X}$, nous posons :

$$p(v) := \min [p(v), p(NODE) + L(NODE,v)]$$

avec la convention que $L(NODE,v)$ égale M si l'arc $(NODE,v)$ n'appartient pas à E et que M + un nombre égale toujours M .

Soit v^* tel que $p(v^*) = \min_{v \text{ dans } \bar{X}} [p(v)]$. Pour tous les arcs $e=(v,v^*)$ dans E pour lesquels $p(v) < p(v^*)$, nous définissons :

$$d(e) := EXP\ \theta L(e) ,$$

$$w(v_o,v^*) := \sum_{v \text{ dans } X} w(v_o,v)\ d(v,v^*) ,$$

$$X := X \cup \{v^*\}$$

et

$$NODE := v^* .$$

Si $Y(v^*)$ égale 0 , nous recommençons à la section *EVALUATION*, sinon nous allons à la section *TEST DE FIN* après avoir augmenté de 1 la variable SUMDEST, c'est-à-dire:

$$SUMDEST := SUMDEST + 1 .$$

TEST DE FIN

Si SUMDEST égale NBDEST, nous poursuivons avec la seconde étape,

BALAYAGE ARRIERE, sinon nous retournons à l'*EVALUATION*.

b) *BALAYAGE ARRIERE*

A ce stade, les voyages sont affectés aux arcs du réseau.

TEST DE FIN

Dénotons par v^* le dernier élément de X. Si v^* égale v_o, nous avons alors terminé et l'affectation est complétée; sinon, nous éliminons le sommet v^* de X et nous passons à la section suivante *EVALUATION*.

EVALUATION

Pour chaque arc $e = (v,v^*)$ dans E tel que $p(v) < p(v^*)$, nous définissons :

$$F(e) := w(v_o,v)\ d(e)\ Y(v^*) \ / \ w(v_o,v^*)$$

et

$$Y[n_1(e)] := Y[n_1(e)] + F(e) .$$

Nous passons à la section *TEST DE FIN*.

Il est opportun à ce moment-ci de faire les remarques suivantes:

- au cours du *BALAYAGE AVANT*, les $p(v)$, les $d(e)$ et les $w(v_o,v)$ sont évaluées simultanément; ce travail nécessite deux balayages du réseau dans l'algorithme initial de Dial,

-au cours du **BALAYAGE ARRIERE**, chaque sommet v rencontré joue le rôle d'une destination où le nombre de voyages entre l'origine v_o et ce même sommet v égale $Y(v)$ + le nombre de voyages passant par v pour se terminer en un autre sommet. Il n'est donc plus nécessaire d'évaluer les fonctions $w(v,v_d)$.

2.3 Justification de l'algorithme

Comme nous l'avons mentionné auparavant, les détails concernant la justification du modèle probabiliste sont présumés connus; aussi présenterons-nous seulement, et brièvement, l'interface entre le modèle et l'algorithme.

Montrons que le nombre approprié de voyages à être affectés à l'arc $e = (v,v^*)$ dans E est donné par l'équation:

$$F(e) = w(v_o,v)\ d(e)\ Y(v^*) \ / \ w(v_o,v^*)$$

apparaissant dans le **BALAYAGE ARRIERE**.

Pour un arc e' dans E , $Pr[e']$ est la probabilité qu'un voyage entre v_o et v_d passe par l'arc e' . Cette même probabilité, sachant que l'on passe par $n_2(e')$, s'exprime ainsi:

$$Pr[e' \mid n_2(e')] = w[v_o,n_1(e')]\ d(e') \ / \ w[v_o,n_2(e')] \qquad (2.6)$$

et

$$Pr[e' \mid n_2(e')] = \frac{Pr[e',n_2(e')]}{Pr[n_2(e')]} = \frac{Pr[e']}{Pr[n_2(e')]} \ . \qquad (2.7)$$

Si $Y_{v_o v_d}$ est le nombre total de voyages commençant en v_o et se terminant en v_d et si $F(e')$ est la proportion de $Y_{v_o v_d}$ utilisant l'arc e', nous obtenons par les équations (2.6) et (2.7) ainsi que par la définition de $Y(e')$:

$$F(e') = Y_{v_o v_d} \Pr[e'] ,$$

$$= Y_{v_o v_d} \Pr[e' \mid n_2(e')] \Pr[n_2(e')] ,$$

$$= Y_{v_o v_d} \Pr[e' \mid n_2(e')] \sum_{e_i \text{ tel que } n_2(e_i) = n_2(e')} \Pr[e_i] ,$$

$$= \Pr[e' \mid n_2(e')] \; Y[n_2(e')]$$

et

$$= w[v_o, n_1(e')] \; d(e') \; Y[n_2(e')] \; / \; w[v_o, n_2(e')] .$$

Puisque chaque sommet $n_2(e')$ peut être considéré comme une destination absorbant $Y[n_2(e')]$ voyages, nous obtenons pour chaque arc $e = (v, v^*)$ dans E :

$$F(e) = \frac{w(v_o, v) \; d(e) \; Y(v^*)}{w(v_o, v^*)} .$$

La généralisation aux cas à plusieurs destinations est directe.

En négligeant les arcs e pour lesquels $p[n_1(e)] > p[n_2(e)]$, nous pouvons démontrer que l'algorithme considère tous les chemins "efficaces" définis par Dial et seulement ces chemins.

2.4 Exemple et résultats numériques

Prenons un exemple et détaillons-le. Soit un réseau formé de 9 sommets et de 14 arcs comme l'illustre la figure 2.1. Les coûts sont indiqués le long des arcs; la demande en voyages entre les sommets 1 et 9 est de 100 et nous supposons que θ égale -1 . Nous précisons les données venant de la distribution des voyages:

$$NBDEST := 1 ,$$

$$Y(i) := 0 \quad \text{pour} \quad i = 1,\ldots, 8$$

et

$$Y(9) := 100 .$$

a) ***BALAYAGE AVANT***

Premièrement, le sommet 1 entre dans $X.X = \{1\}$. La section *INITIALISATION* se ramène aux opérations:

$$p(i) := M \quad \text{pour} \quad i = 1,\ldots, 9 ,$$

$$F(k) := 0 \quad \text{pour} \quad k = 1,\ldots, 14 ,$$

$$NODE := 1 ,$$

$$SUMDEST := 0$$

et

$$p(1) := 0 .$$

Nous passons ensuite à la section *EVALUATION*. Pour tous les sommets v dans $\bar{X}$ pour lesquels (1,v) appartient à E , nous calculons de nouvelles pondérations:

$$p(2) := 2 \quad ,$$

$$p(4) := 2$$

et

$$p(5) := 3 \quad .$$

La pondération des autres sommets demeure inchangée. Nous prenons la valeur minimale des $p(i)$ pour lesquelles i appartient à $\bar{X}$. Le minimum est 2 pour les sommets 2 et 4. Choisissons le sommet 2. Ce choix nous amène, en respectant la contrainte reliée à la notion de chemin "efficace", à calculer les valeurs:

$$d(1,2) := EXP\ (-1 * 2) = 0.135$$

et

$$w(1,2) := w(1,1)\ d(1,2) = 0.135 \quad .$$

L'ensemble X est alors égal à $\{1,2\}$. Du sommet 2, nous calculons $p(3) := 4$, $p(5) := 3$ et nous prenons la valeur minimale des $p(i)$ pour lesquelles i appartient à $\bar{X}$, soit le sommet 4. Nous calculons $d(v,4)$ pour tous les arcs $(v,4)$ dans E pour lesquels $p(v) < p(4)$:

$$d(1,4) := EXP\ (-1 * 2) = 0.135 \quad .$$

De plus, nous calculons:

$$w(1,4) := w(1,1)\ d(1,4) = 0.135 \quad .$$

Le processus se poursuit jusqu'à ce que le sommet-destination 9 soit atteint. Les valeurs $p(i)$, $d(k)$ et $w(1,i)$ obtenues au terme du ***BALAYAGE AVANT*** apparaissent à la figure 2.2 et l'ensemble X se lit comme suit: $X = \{1,2,4,5,7,3,6,8,9\}$.

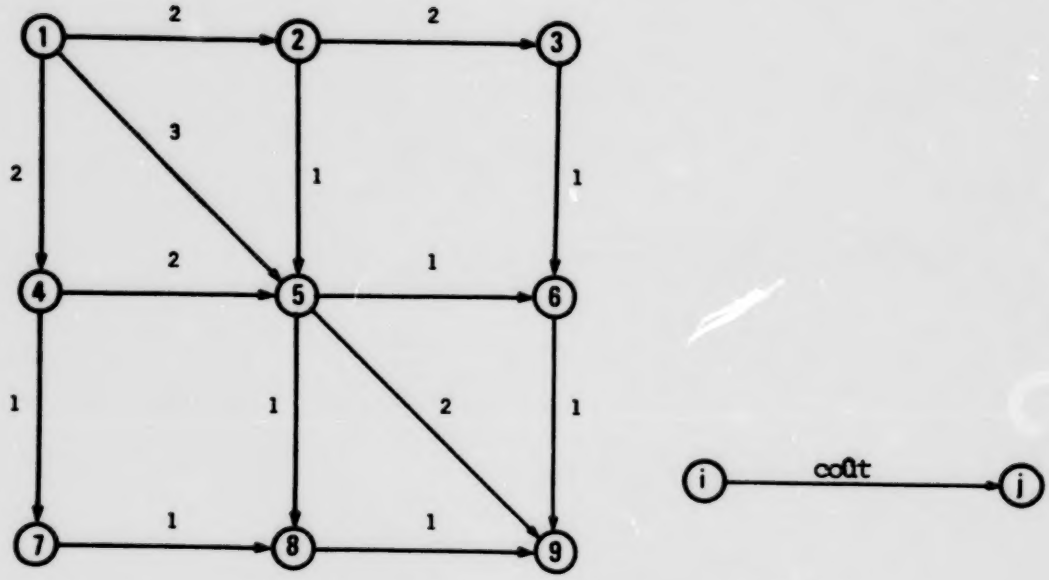


figure 2.1 Réseau exemplaire avec ses coûts.

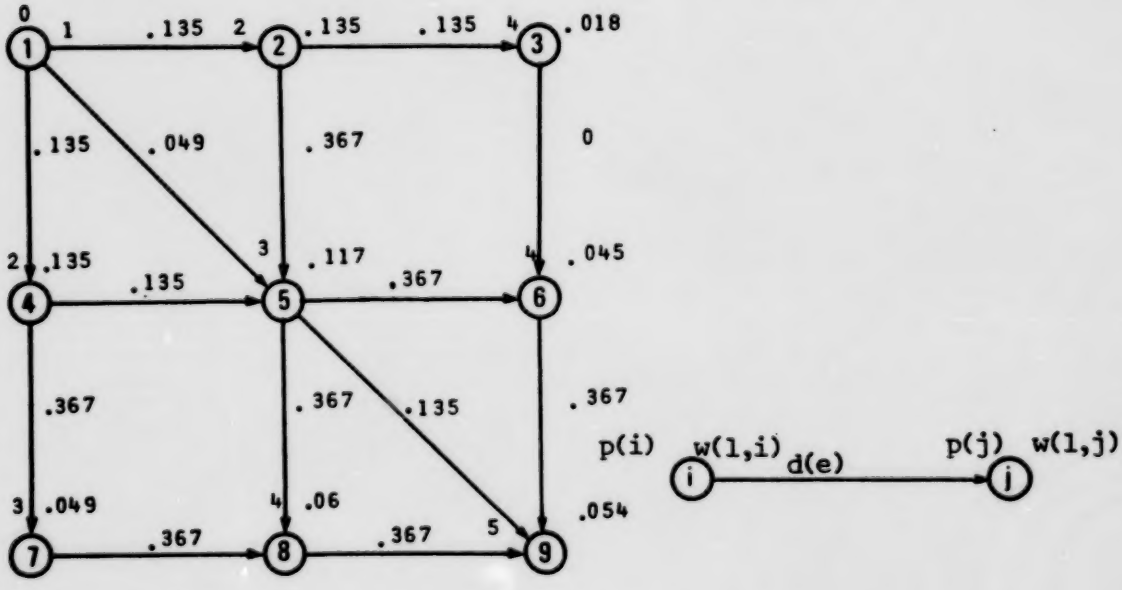


figure 2.2 Réseau exemplaire avec les valeurs obtenues au terme du *BALAYAGE AVANT*.

b) **BALAYAGE ARRIERE**

Le sommet 9 est éliminé de X . Partant du sommet 9, la section *EVALUATION* nous amène à calculer les valeurs $F(i,9)$ et $Y(i)$ pour les sommets i tels que $p(i)$ est plus petit que $p(9)$ et que l'arc $(i,9)$ appartient à E :

$$F(6,9) := \frac{w(1,6)\ d(6,9)\ Y(9)}{w(1,9)} = \frac{0.045 * 0.367 * 100}{0.054} = 29.2 ,$$

$$Y(6) := 29.2 ,$$

$$F(5,9) := \frac{w(1,5)\ d(5,9)\ Y(9)}{w(1,9)} = \frac{0.117 * 0.135 * 100}{0.054} = 29.2 ,$$

$$Y(5) := 29.2 ,$$

$$F(8,9) := \frac{w(1,8)\ d(8,9)\ Y(9)}{w(1,9)} = \frac{0.06 * 0.367 * 100}{0.054} = 41.5$$

et

$$Y(8) := 41.5 .$$

Les valeurs correspondantes au sommet 8 sont:

$$F(5,8) := \frac{w(1,5)\ d(5,8)\ Y(8)}{w(1,8)} = \frac{0.117 * 0.367 * 41.5}{0.061} = 29.2 ,$$

$$Y(5) := 29.2 + 29.2 = 58.4 ,$$

$$F(7,8) := \frac{w(1,7)\ d(7,8)\ Y(8)}{w(1,8)} = \frac{0.049 * 0.367 * 41.5}{0.061} = 12.3$$

et

$$Y(7) := 12.3 .$$

Tous les sommets sont balayés de la même façon. Lorsque le sommet 1 est atteint, l'affectation est réalisée. La figure 2.3 donne les valeurs finales.

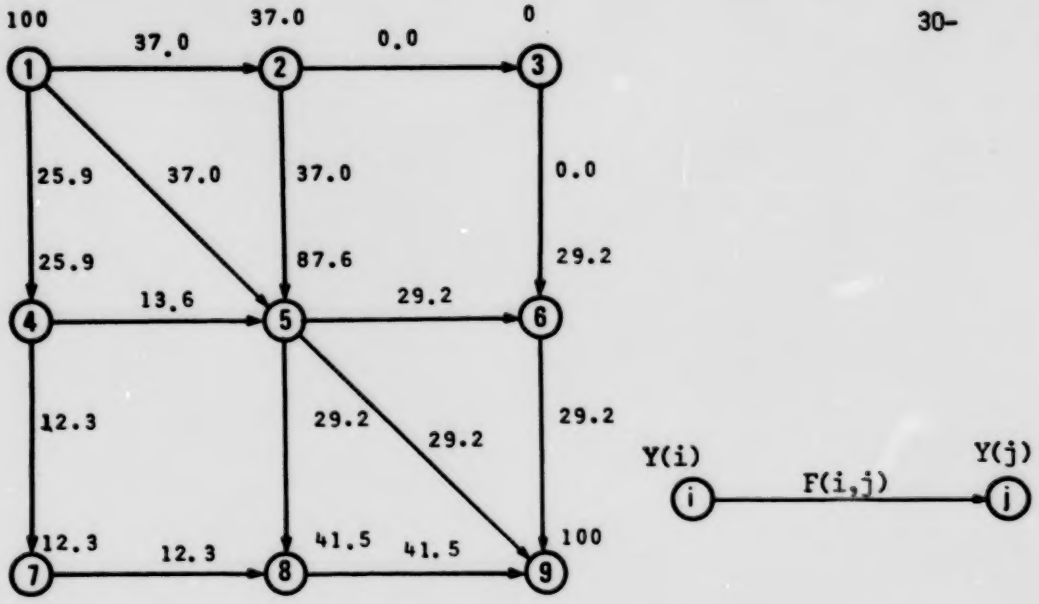


figure 2.3 Réseau exemplaire avec sa charge finale.

Nous avons programmé cet algorithme en FORTRAN IV. La mise au point du programme a été faite sur l'ordinateur CDC CYBER 74 du Centre de Calcul de l'Université de Montréal; cet ordinateur a une capacité-mémoire de 140000_8 mots de 60 bits chacun. Le programme peut traiter des réseaux ayant jusqu'à 7000 éléments (sommets et arcs). Il n'utilise pas de segments de recouvrement (overlay) et tous les exemples présentés nécessitent moins de 60000_8 mots-mémoire.

L'exemple 1 correspond à celui que nous avons étudié précédemment. L'exemple 2 traite un réseau extrait de l'exemple 1, soit un réseau composé des quatre premiers sommets du réseau inhérent à l'exemple 1. Les exemples 3 et 7 ont été proposés par Dial (1971), et les exemples 4, 5, 6, 8 et 9 sont des variantes des deux précédents. Le réseau associé à l'exemple 10 représente les routes principales de la ville de Hull au Québec, Canada.

No Ex.	(sommets	Nombre Arcs	Orig.	Dest. par orig.)	Temps pour effectuer l'affectation d'une orig.	Temps pour effectuer l'affectation totale	θ
1	9	14	1	1	.004 s.	.004 s.	-1.0
2	4	5	1	1	.001	.001	-1.0
3	9	32	1	1	.004	.004	-1.0
4	9	32	1	8	.006	.006	-1.0
5	9	32	3	3	.004	.013	-1.0
6	25	40	1	1	.014	.014	-1.0
7	25	80	1	1	.015	.015	-1.0
8	25	80	1	3	.019	.019	-1.0
9	25	80	4	3	.017	.060	-1.0
10	155	377	27	27	.34	≈10.0	-0.1

2.5 Performance de l'algorithme

L'algorithme proposé diffère de celui de Dial en plusieurs points. Nous allons en discuter quelques-uns.

L'étape 0 (préliminaire) apparaissant dans l'algorithme de Dial est évitée. Les valeurs d(e) et p(v) qui étaient alors calculées, le sont implicitement durant le *BALAYAGE AVANT*. Nous parcourons donc le réseau une fois de moins. Il est à noter que le nombre de d(e) évaluées est inférieur (ou égal) au nombre d'arcs et que le nombre de p(v) évaluées est inférieur (ou égal) au nombre de sommets dans le réseau.

D'autre part, le nombre de valeurs à calculer est réduit: nous savons qu'un grand nombre de d(e) , de $w(v_i,v_j)$ et de Pr[e] restent à zéro par rapport à une origine, aussi l'algorithme proposé tente de ne

calculer que les valeurs qui sont possiblement différentes de zéro.

Cet algorithme simplifie également la notion d'utilité associée aux arcs du réseau. En effet, la fonction d(e) représente l'utilité relative de l'arc e ; elle est équivalente à la fonction a(e) dans l'algorithme de Dial et ne dépend maintenant que de la longueur de l'arc considéré, soit L(e) .

L'algorithme améliore la façon de pondérer les éléments du réseau: les pondérations sont associées aux sommets du réseau au lieu d'être associées aux arcs. Puisque dans la plupart des cas le nombre de sommets est inférieur au nombre d'arcs, le nombre des valeurs à calculer et à manipuler est moindre. En termes d'ordinateur, nous épargnons donc temps et espace.

Enfin, l'algorithme proposé simplifie les différentes formules à évaluer. Dans le tableau suivant, nous comparons les formules apparaissant dans les deux algorithmes.

ALGORITHME DE DIAL

$$a(e) = \begin{cases} \text{EXP } \theta\{p[n_2(e)] - p[n_1(e)] - L(e)\} & \text{si } p[n_1(e)] < p[n_2(e)] \\ 0 & \text{autrement} \end{cases}$$

$$w(e) = \begin{cases} a(e) & \text{si } n_1(e) = v_o \\ a(e) \sum_{e' \text{ tel que } n_2(e') = n_1(e)} w(e') & \text{autrement} \end{cases}$$

$$F(e) = \frac{w(e)\, Y[n_2(e)]}{\sum_{e' \text{ tel que } n_2(e') = n_2(e)} w(e')}$$

ALGORITHME PROPOSE

$$d(e) = \text{EXP } \theta L(e)$$

$$w(v_o, v_o) = 1$$

$$w[v_o, n_2(e)] = \sum_{v \text{ tel que } p(v) < p[n_2(e)]} w(v_o, v)\, d[v, n_2(e)]$$

$$F(e) = \frac{w[v_o, n_1(e)]\, d(e)\, Y[n_2(e)]}{w[v_o, n_2(e)]}$$

Dans tous les cas, l'algorithme proposé donne lieu à des formules plus simples à évaluer.

CHAPITRE III

UN MODELE D'AFFECTATION DYNAMIQUE

3.1 Historique et commentaires

Ce chapitre généralise le modèle d'affectation probabiliste introduit au chapitre précédent aux cas où la demande en transport varie avec le temps, où les temps de parcours et les vitesses sur les sections de chemin sont des variables aléatoires et où les éléments qui composent le critère de dispersion des voyageurs sur les différents chemins possibles varient dans le temps. Les modèles d'affectation utilisés jusqu'à ce jour sont liés à des hypothèses très strictes; en effet, ils considèrent la demande en transport, les temps de parcours, les vitesses et les éléments constituant le critère de dispersion comme étant constants dans le temps. De plus, tous ces modèles calculent pour chacune des sections de chemin un seul nombre qui représente leur charge. Très vite, nous nous rendons compte que leurs domaines d'application sont limités: nous pouvons étudier le comportement des voyageurs à des moments précis, par exemple, durant les périodes stables de la journée en dehors des heures de pointe; l'hypothèse de l'état stationnaire de la demande a alors un sens et le flot des arrivées correspond à une moyenne faite à partir des observations prises durant cette période. Ces modèles permettent également d'étudier le comportement des voyageurs ainsi que l'efficacité du réseau de transport aux moments critiques en

prenant pour flot constant, le flot maximal observé aux heures de pointe ou encore la moyenne des observations notées à ces heures. Dans certains cas, l'analyse simple et spécialisée que ces modèles permettent est en première approximation une bonne représentation de la réalité. Il importe néanmoins de développer des modèles permettant une application plus vaste, à savoir une étude dans le temps du comportement des voyageurs.

Les hypothèses introduites dans ce chapitre permettront une étude dynamique de la répartition du trafic, la prédiction des heures de pointe aux différents endroits du réseau ainsi que la mesure de leur durée. Comme l'ont noté Huber, Boutwell et Withford (1968), la généralisation des modèles classiques d'affectation aux cas dynamiques serait une étape importante pour une meilleure compréhension du phénomène du transport.

Le modèle d'affectation proposé calcule pour chaque section de chemin non plus un seul nombre, mais la répartition du trafic en fonction du temps. L'effort de calcul pour l'obtention des courbes "volume vs temps" pour chaque section de chemin est évidemment supérieur à celui requis dans les modèles conventionnels, mais il demeure toutefois dans des limites réalistes.

Les développements exposés ont comme point de départ le modèle d'affectation probabiliste proposé par Dial (1971) et implanté sous le nom de STOCH (1972). Depuis sa publication, cette approche de Dial a suscité maintes controverses dont les arguments principaux sont discutés par Schneider (1973). En résumé, le modèle d'affectation de Dial se veut une généralisation de celui du "tout ou rien" et, comme ce dernier, il ne permet pas dans une seule itération de considérer les fonctions sur les arcs comme dépendant du volume. Cependant, nous pouvons simuler

cette dépendance en utilisant un processus itératif où les fonctions sur les arcs sont ajustées à partir des volumes obtenus lors d'affectations précédentes. Pour juger de l'efficacité des méthodes itératives, nous nous référons à la publication de Huber, Boutwell et de Withford (1968) qui présente une étude comparative empirique de différentes méthodes itératives. Les conclusions de cette étude, quoique incomplètes et limitées, sont favorables à l'utilisation des méthodes itératives pour reproduire l'effet de la congestion.

Ce chapitre s'attarde essentiellement au développement d'une itération du processus itératif. Aussi, comme les fonctions sur les arcs du réseau dépendent des répartitions du volume issues des itérations précédentes et non du volume qu'on affecte, les fonctions sur les arcs sont fixes pour la présentation qui suit.

3.2 Avertissement

Ce travail a pour but premier de présenter un modèle d'un point de vue théorique. Il s'ensuit que très peu de précisions sont données, et cela volontairement, sur les éléments qui constituent le réseau de façon générale et pour lesquels la théorie énoncée s'applique. Il revient à chacun de juger de la pertinence de son propre réseau, étant donné les exigences imposées par le modèle même. Cependant, on peut convenir que le réseau doit être composé de sections de chemins qui peuvent être groupées de telle sorte que les éléments des groupes soient comparables: cette comparaison s'établit au niveau de leur habileté à satisfaire une même demande en trafic. De plus, les sections de che-

mins doivent être telles que la vitesse des véhicules sur ces éléments du réseau soit le facteur dominant pour juger du niveau de service; en outre, la longueur de ces sections doit être assez grande pour négliger les interférences causées par les piétons, les temps d'attente, les virages, aux intersections par exemple. Sur les voies rapides modernes construites selon des normes uniformes, ces conditions sont généralement respectées; nous pouvons considérer comme telles les voies allant des artères principales en milieu urbain aux autoroutes en milieu rural. Les autres voies ne sont pas à exclure définitivement, mais il faudrait spécialiser le modèle préconisé ici pour les y intégrer.

3.3 Notation, définitions et exigences

Avant d'entreprendre pour de bon la discussion des différents aspects de la théorie, nous nous devons de préciser certains éléments. Ainsi, un réseau de transport est dorénavant représenté par un graphe G décrit à l'aide d'un quadruplet: $G = [V, E, n, d_{v_o}]$ où d_{v_o} est une généralisation de la fonction d rencontrée au chapitre précédent. d_{v_o} associe à chaque arc e dans E, à chaque origine v_o ainsi qu'à chaque temps de départ T de v_o, un nombre non négatif noté $d_{v_o}(e,T)$. La fonction d_{v_o} mesure l'utilité de chaque arc du réseau comme la perçoit un voyageur quittant l'origine v_o au temps T. Si e n'appartient pas à E, $d_{v_o}(e,T)$ égale alors 0. En vue de simplifier la présentation, nous supposons que le réseau possède une seule origine v_o

et une seule destination v_d. Cette restriction pourra être abandonnée sans complication. Dans ce contexte, d_{v_o} est notée d.

A partir du réseau de transport, nous extrayons un graphe G et nous exigeons qu'il ne contienne pas de circuits orientés. Comme nous l'avons déjà mentionné, cette restriction est nécessaire pour permettre de calculer récursivement les probabilités sur les éléments du réseau, mais en réalité, les usagers n'empruntent pas les chemins possédant des circuits; or, cette restriction ne vient que renforcer le modèle d'affectation. En outre, cette restriction assure que les usagers ont à choisir entre un nombre fini de chemins puisque V et E sont finis, ce qui est nécessaire pour que les mesures de probabilité aient un sens. Enfin, nous exigeons que tous les chemins de v_o à v_d aient une probabilité non nulle d'être choisis et que tous les éléments de G appartiennent à un chemin allant de v_o à v_d.

3.4 Modèle d'affectation dynamique

La présentation de l'aspect dynamique du modèle requiert l'introduction des définitions suivantes:

T est le temps de départ d'un usager de l'origine v_o ;

f_o est la fonction de densité de la variable aléatoire T ;

$t_{e|n}$ est le temps requis à un usager partant de v_o au temps n (c'est-à-dire $T = n$) pour parcourir l'arc e ; nous supposons que les $t_{e|n}$ sont indépendamment distribuées;

$g_{e|n}$ est la fonction de densité de $t_{e|n}$;

$t_{p|n}$ est le temps nécessaire à un usager partant de v_o au temps n pour parcourir le chemin p. Lorsqu'un usager a commencé son voyage, il ne peut attendre en cours de route que la congestion ait diminué pour repartir. Cette situation s'exprime ainsi:

$$t_{p|n} = \sum_{E} \delta_{ep} \, t_{e|n} \quad ;$$

$g_{p|n}$ est la fonction de densité de $t_{p|n}$.

$Pr(p|n)$ pour p dans P_v est la probabilité qu'un voyageur partant de v_o au temps n emprunte le chemin p sachant qu'il doit se rendre au sommet v. Conformément au modèle d'affectation dans le cas stationnaire, nous définissons cette probabilité ainsi:

$$Pr[p|n] = K \prod_{E} d(e,n)^{\delta_{ep}} \quad \text{avec} \quad p \quad \text{dans} \quad P_v$$

où K est une constante de normalisation.

Pour les usagers qui atteignent le sommet v, v étant différent de v_o, nous appelons t_v le temps d'arrivée au sommet v. f_v est la fonction de densité de t_v .

Pour les usagers qui atteignent le sommet v, v étant différent de v_o, et dont le temps de départ de v_o est n, nous définissons $t_{v|n}$ comme le temps d'arrivée au sommet v et $f_{v|n}$ comme la fonction de densité de $t_{v|n}$.

La probabilité d'être sur l'arc e au temps t, sachant qu'on va de v_o à v_d, est dénotée par $Pr(e,t)$.

Par analogie avec le cas stationnaire, nous définissons la probabilité qu'un usager partant de l'origine au temps n traverse l'arc $e = (v_i,v_j)$ dans E par :

$$Pr[e|n] = \sum_E \delta_{ep} \, Pr(p|n) \quad \text{avec } p \text{ dans } P .$$

Substituant dans cette dernière équation l'expression définissant $Pr(p|n)$, on peut écrire:

$$Pr[e|n] = \frac{w(v_o,v_i,n)\, d(e,n)\, w(v_j,v_d,n)}{w(v_o,v_d,n)}$$

où

$$w(v_i,v_j,n) = \sum_{P_{v_i v_j}} \prod_E d(e,n)^{\delta_{ep}}$$

en convenant que $w(v_i,v_j,n)$ égale 0 lorsque $P_{v_i v_j}$ est un ensemble vide, et égale 1 lorsque i égale j. La sommation est sur l'ensemble des chemins allant de v_i à v_j. Les quantités $w(\cdot,\cdot,\cdot)$ peuvent être évaluées de façon récursive:

$$w(v_i,v_j,n) = \sum_V w(v_i,v,n)\, d[(v,v_j),n]$$

ou encore:

$$w(v_i,v_j,n) = \sum_V w(v,v_j,n)\, d[(v_i,v),n]$$

pour tous les sommets v_i,v_j dans V et pour un temps de départ n du

sommet v_o .

Le calcul des $w(\cdot,\cdot,\cdot)$ peut donc être réalisé de façon extrêmement efficace. Nous tâcherons d'ailleurs d'exprimer toutes les formules à l'aide de ces quantités.

3.5 Evaluation des fonctions de densité

Nous sommes maintenant en mesure d'exprimer f_v, $f_{v|n}$ et $Pr(e,t)$ en fonction de f_o , $g_{e|n}$ et $Pr(p|n)$ qui sont les données du problème.

On a d'abord:

$$f_v(x) = \int_0^\infty f_o(n \mid \text{passe par } v)\ f_{v|n}(x)\ dn$$

où $f_o(n \mid \text{passe par } v)$ est la fonction de densité associée à la variable aléatoire indiquant le temps de départ de v_o sachant qu'on passe par v pour se rendre en v_d .

On a alors:

$$f_v(x) = \int_0^\infty f_o(n)\ \frac{Pr(v|n)}{Pr(v)}\ f_{v|n}(x)\ dn \ .$$

$Pr(v)$ égale $\int_0^\infty Pr(v|n)\ f_o(n)\ dn$ et $Pr(v|n)$ est la probabilité de passer par le sommet v étant donné qu'on quitte v_o au temps n pour se rendre en v_d . Par hypothèse, nous savons que $Pr(v) > 0$. Il nous suffira donc de pouvoir évaluer $f_{v|n}$ pour connaître f_v . A cette fin, conditionnons $f_{v|n}$ par rapport au fait qu'on choisisse le chemin p pour aller de v_o à v .

$$f_{v|n}(x) = \sum_{P_v} Pr(p|n)\ f_{v|n}(x \mid \text{utilise } p)$$

où $f_{v|n}(x \mid \text{utilise } p)$ est la fonction de densité associée à la variable aléatoire indiquant le temps d'arrivée en v pour un usager utilisant le chemin p dans P_v et partant de v_o au temps n.

On en déduit que:

$$f_{v|n}(x) = \sum_{P_v} Pr(p|n)\ g_{p|n}(x-n) \quad .$$

Notons que la sommation se fait sur l'ensemble des chemins entre v_o et v et que $Pr(p|n)$ est évaluée seulement sur l'ensemble restreint qu'est P_v.

L'évaluation efficace de $f_{v|n}$ dépend de la facilité avec laquelle cette sommation peut être calculée sans énumération des chemins entre v_o et v. Pour ce faire, nous considérons la fonction caractéristique de $f_{v|n}$, soit $\Phi_{v|n}$.

$$\Phi_{v|n}(Z) = \int_{-\infty}^{\infty} \exp(iZx)\ f_{v|n}(x)\ dx \ .$$

Comme $g_{p|n}$ est la fonction de densité d'une variable aléatoire qui est elle-même une somme de variables aléatoires indépendantes, on vérifie que $\Phi_{v|n}$ s'écrit:

$$\Phi_{v|n}(Z) = \frac{\exp(iZn) \sum_{P_v} \prod_{E} [d(e,n)\ \Phi_{e|n}(Z)]^{\delta_{ep}}}{w(v_o,v,n)}$$

où $\Phi_{e|n}$ est la fonction caractéristique de $t_{e|n}$.

Soulignons le fait que le terme au numérateur est identique à une fonction $w(\cdot,\cdot,\cdot)$ et se calcule de la même façon. Si on pose:

$$w_Z(v_o,v,n) = \sum_{P_v} \prod_E [d(e,n)\ \Phi_{e|n}(Z)]^{\delta_{ep}}$$

par analogie avec

$$w(v_o,v,n) = \sum_{P_v} \prod_E d(e,n)^{\delta_{ep}} ,$$

on obtient:

$$\Phi_{v|n}(Z) = \frac{\exp(iZn)\ w_Z(v_o,v,n)}{w(v_o,v,n)} .$$

Nous pouvons donc évaluer aisément la fonction caractéristique de $f_{v|n}$ pour chaque point Z désiré. Utilisant les formules numériques d'inversion des fonctions caractéristiques décrites par Bohmann (1970, 1972) ou par Davies (1973), on peut calculer la fonction $f_{v|n}$. Le seul terme nouveau est $\Phi_{e|n}$.

La définition des fonctions des temps de parcours relève essentiellement du domaine expérimental. Le prélèvement de ces données peut s'effectuer de plusieurs façons. Analysons-en une:

Le Highway Research Board (1965) et le Department of Scientific and Industrial Research, Road Research Laboratory (1965) affirment qu'en général la vitesse des véhicules suit une distribution normale (μ,σ^2) où $\sigma = .2 * \mu$. Cette affirmation permet de considérer que les vitesses

sur chaque arc e pour un temps de départ du sommet v_o égal à n sont distribuées selon une normale $(\mu'_{e|n}, 0.04 * \mu'^{2}_{e|n})$, où $\mu'_{e|n}$ est la vitesse moyenne d'un véhicule sur l'arc e issu de v_o au temps n. En considérant la normale $(\mu'_{e|n}, 0.04 * \mu'^{2}_{e|n})$, on peut montrer qu'une bonne approximation de la fonction de distribution de $t_{e|n}$ est une normale $\left[\frac{L(e)}{\mu'_{e|n}}, \left(.2 * \frac{L(e)}{\mu'_{e|n}}\right)^2\right]$ où L(e) est la longueur de l'arc e.

Donc, pour un arc e et pour un temps de départ égal à n, nous avons les relations suivantes:

$V_{e|n}$, la vitesse d'un véhicule sur l'arc e s'il part de v_o au temps n, est distribuée suivant une normale $\left[\mu'_{e|n}, (.2 * \mu'_{e|n})^2\right]$;

$t_{e|n}$ est approximativement distribuée suivant une normale

$$(\mu_{e|n}, \sigma^2_{e|n}) \quad \text{où} \quad \mu_{e|n} = \frac{L(e)}{\mu'_{e|n}} \quad \text{et} \quad \sigma_{e|n} = \frac{.2 * L(e)}{\mu'_{e|n}} .$$

$\Phi_{e|n}$, apparaissant dans l'expression de $\Phi_{v|n}$, s'obtient donc de façon explicite:

$$\Phi_{e|n}(Z) = \exp(iZ\mu_{e|n} - \frac{Z^2\sigma^2_{e|n}}{2}) .$$

Une autre fonction intéressante est Pr(e,t). En effet, s'il y a $Y_{v_o v_d}$ usagers allant de v_o à v_d durant l'intervalle de temps

sous étude, le nombre d'usagers sur l'arc e au temps t est égal à $Y_{v_o v_d}\ Pr(e,t)$. Aussi, le problème d'affectation dans le cas dynamique revient à l'évaluation des $Pr(e,t)$ pour tout e dans E à différents temps.

Soit $e = (k,\ell)$. En conditionnant $Pr(e,t)$ par rapport au temps de départ de v_o, on obtient:

$$Pr(e,t) = \int_0^\infty f_o(n)\ Pr(e,t|n)\ dn \ .$$

$Pr(e,t|n)$ est la probabilité qu'un usager partant de v_o au temps n soit sur l'arc e au temps t .

Si $Pr[e,t|n,\text{passe par } k]$ est la probabilité qu'un usager partant de v_o au temps n et passant par k soit sur l'arc e au temps t , on peut écrire:

$$Pr(e,t) = \int_0^\infty f_o(n)\ Pr[e,t|n,\text{ passe en } k]\ Pr[k|n]\ dn \ .$$

Nous conditionnons $Pr[e,t|n,\text{ passe en } k]$ suivant le fait que l'usager passe en k au temps x .

$$Pr(e,t) = \int_0^\infty f_o(n)\ Pr[k|n] \int_0^t Pr[e,t|n,\text{ passe en } k \text{ au temps } x]\ f_{k|n}(x)\ dx\ dn \ .$$

Si l'usager part de v_o au temps n et qu'il arrive au sommet k au temps x , $Pr(e,t|n,\text{ en } k \text{ au temps } x)$ est alors la probabilité qu'il soit sur l'arc e au temps t . D'autre part, il est facile de vérifier que:

$Pr[e,\ t|n,\text{ passe en } k \text{ au temps } x] = Pr[t_{e|n} > t-x]\ Pr(e|\ n,\ k)$

où $Pr(e|n,\ k)$ est la probabilité qu'un voyageur emprunte l'arc e s'il part de v_o au temps n et qu'il passe par le sommet k .

Donc,

$$Pr(e,t) = \int_0^{\infty} f_o(n)\, Pr(e|n) \int_0^{t} Pr(t_{e|n} > t - x)\; f_{k|n}(x)\, dx\, dn \quad . \tag{3.1}$$

Cette dernière formule est appelée formule d'affectation dans le cas dynamique. Il est à noter que tous les termes qui forment cette formule peuvent être évalués de façon efficace: $f_o(n)$ est donnée; $Pr(e|n)$ revient au problème d'affectation dans le cas stationnaire; $Pr(t_{e|n} > t - x)$ s'obtient directement à l'aide de la cumulative d'une normale (0,1) puisque $t_{e|n}$ suit une loi de distribution normale; $f_{k|n}$ a été traitée précédemment. Le problème vient donc de l'évaluation globale de l'expression, principalement à cause des intégrales.

3.6 Partition des temps de départ

La formule (3.1) donne la forme théorique de la formule d'affectation. En pratique, cette formule est difficile à manipuler principalement à cause des intégrales et de la "généralité" de la fonction f_o. Nous allons énoncer des hypothèses supplémentaires sur le modèle et discuter des conséquences qu'elles ont sur l'évaluation des $Pr[e,t]$.

Supposons qu'on étudie l'affectation sur le réseau durant l'intervalle $[a,b)$, c'est-à-dire que les temps de départ des voyageurs en v_o appartiennent à cet intervalle. Nous émettons l'hypothèse que cet intervalle peut être séparé de sorte que pour tout sous-intervalle $[a_j,a_{j+1})$ où $j = 1,\ldots,r$, où $a_1 = a$ et où $a_{r+1} = b$, un usager perçoit le réseau de façon identique, que son temps de départ s'inscri-

ve au début ou à la fin du sous-intervalle et de sorte que la distribution des départs soit uniforme sur le sous-intervalle. Cette hypothèse implique que les $g_{e|n}$, la $f_o(n)$ et les $d(e,n)$ sont des constantes pour tout n appartenant au même sous-intervalle. Puisque le nombre de ces sous-intervalles et leur longueur sont facultatifs, cette hypothèse ne constitue pas une véritable contrainte.

Nous pouvons résumer la situation en écrivant:

$$[a,b) = [a_1,a_2) \text{ -- } [a_j,a_{j+1}) \text{ -- } [a_r,a_{r+1})$$

de sorte que

$d(e,n)$ est une constante et égale $d(e,j)$ pour n appartenant à $[a_j,a_{j+1})$;

$f_o(n)$ est égale à $f_o(j)$ pour n appartenant à $[a_j,a_{j+1})$;

$g_{e|n}$ est une fonction de densité connue et égale $g_{e|j}$ pour n appartenant à $[a_j,a_{j+1})$, ce qui implique que

$Pr(p|n)$ est une constante sur l'intervalle $[a_j,a_{j+1})$ et égale $Pr(p|j)$.

La formule d'affectation devient:

$$Pr(e,t) = \sum_{j=1}^{r} f_o(j)\, Pr(e|j) \int_0^t Pr(t_{e|j} > t - x) \int_{a_j}^{a_{j+1}} f_{v|n}(x)\, dn\, dx$$

et la fonction caractéristique de $f_{v|n}$ se ramène à:

$$\Phi_{v|n}(Z) = \frac{\exp(iZn)\, w_Z(v_o,v,j)}{w(v_o,v,j)} \quad .$$

Pour évaluer la formule d'affectation, il faut calculer numériquement deux intégrales. Dans un contexte général, le problème est d'approcher numériquement l'intégrale:

$$I(f) = \int_a^b f(x)\, dx \quad a \text{ et } b \text{ dans } \mathbb{R}$$

par

$$L(f) = \sum_{j=1}^{m} S_j\, f(x_j)\quad x_j \text{ dans } [a,b] \quad .$$

On définit l'erreur par le terme $E(f)$:

$$E(f) = | I(f) - L(f) | \quad .$$

Il faut déterminer le choix d'abscisses x_j et de poids S_j pour lequel l'erreur est minimale pour un m fixe. Il est connu que les fonctions splines approchent de façon "douce" un ensemble de données et qu'elles offrent un haut degré de versatilité dans les problèmes particuliers qui nous concernent. Un article de T.N.E. Greville (1967) présente la théorie relative aux fonctions splines et de nombreux algorithmes sont disponibles pour évaluer numériquement des intégrales à l'aide des fonctions splines. Ces algorithmes demandent très peu d'opérations.

3.7 Etude de la moyenne et de la variance de f_v

Le but de cette section est d'étudier la moyenne et la variance de f_v en utilisant la formule qui relie les moments et la fonction caractéristique, soit:

$$n^{\text{ème}} \text{ moment} = \left(\frac{1}{i}\right)^n \left. \frac{d^n \Phi_v(Z)}{dZ^n} \right|_{Z=0} .$$

a) La moyenne de f_v

Soit μ_v la moyenne de f_v. Dans la formule des moments, en substituant à Φ_v une expression similaire à celle obtenue pour $\Phi_{v|n}$, nous obtenons:

$$\mu_v = \int_0^\infty \frac{n\, f_o(n)\, Pr(v|n)}{Pr(v)}\, d_n + \int_0^\infty f_o(n)\, \frac{Pr(v|n)}{Pr(v)} \sum_{P_v} \mu_{p|n}\, Pr(p|n)\, dn$$

$$\mu_v = \mu_{o|v} + \mu .$$

En partant du fait que $\mu_{p|n}$ est le temps moyen pour franchir le chemin p lorsque l'usager part de v_o au temps n, l'expression précédente peut se lire ainsi: μ_v, la moyenne de f_v, égale $\mu_{o|n}$ qui représente le temps de départ moyen pour passer par v auquel temps s'additionne μ qui est le temps moyen pour se rendre en v pour l'usager passant par v.

Suivant les hypothèses énoncées précédemment, nous obtenons:

$$\mu_v = \sum_{j=1}^{r} f_o(j) \frac{Pr(v|j)}{Pr(v)} \int_{a_j}^{a_{j+1}} ndn + \sum_{j=1}^{r} f_o(j) \frac{Pr(v|j)}{Pr(v)} \sum_{P_v} Pr(p|j)\, \mu_{p|j} \int_{a_j}^{a_{j+1}} dn \ .$$

Posons $A_j = f_o(j) \frac{Pr(v|j)}{Pr(v)}$ et $b_j = a_{j+1} - a_j$.

Nous pouvons alors écrire:

$$\mu_v = \sum_{j=1}^{r} A_j \frac{b_j^2}{2} + \sum_{j=1}^{r} A_j b_j \sum_{E} \mu_{e|j}\ Pr(e|j) \ .$$

Ces termes peuvent être obtenus de la façon habituelle.

b) La variance de f_v

En utilisant la formule des moments et celle qui relie la variance, la moyenne et le deuxième moment, nous obtenons que σ_v^2 , la variance de f_v , est telle que:

$$\sigma_v^2 = \sum_{j=1}^{r} \frac{A_j b_j^3}{3} + \sum_{j=1}^{r} b_j C_j + \sum_{j=1}^{r} A_j b_j \sum_{E} Pr(e|j)\ [\sigma_{e|j}^2 + \mu_{e|j}^2]$$

$$+ \sum_{j=1}^{r} A_j b_j \sum_{P_v} Pr(p|j)\ D(p,j) - \left(\sum_{j=1}^{r} \frac{A_j b_j}{2} \right)^2 - \left(\sum_{j=1}^{r} C_j \right)^2$$

$$- \left(\sum_{j=1}^{r} A_j b_j^2 \right) \left(\sum_{j=1}^{r} C_j \right)$$

où

$$C_j = A_j b_j \sum_E \Pr(e|j)\, \mu_{e|j}$$

et

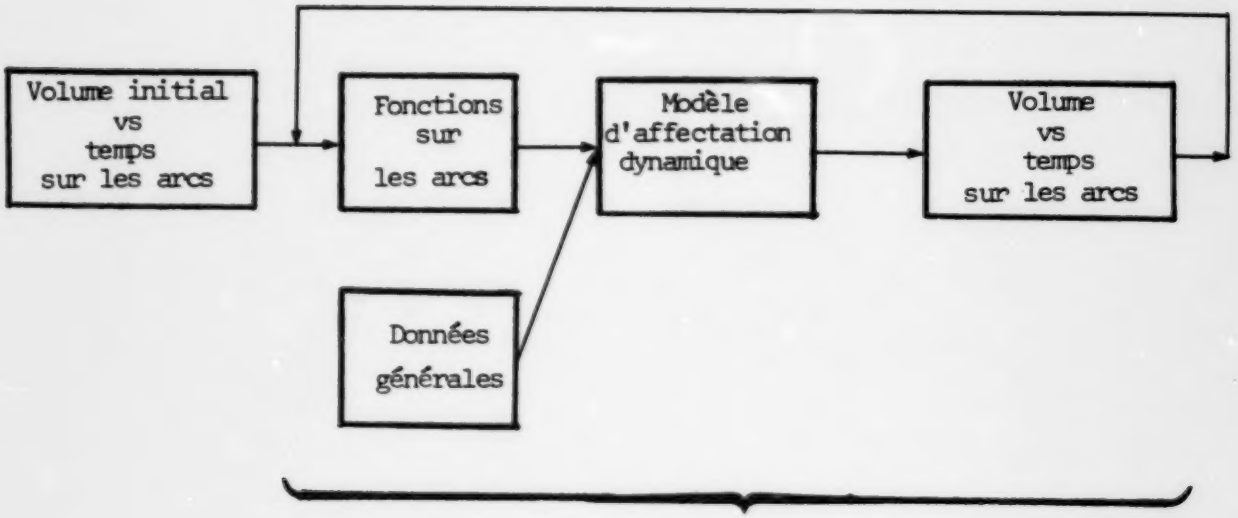
$$D(p,n) = \sum_E \sum_{e' \neq e \text{ dans } E} \delta_{ep} \delta_{e'p}\, \mu_{e|n}\, \mu_{e'|n} \quad .$$

Comme prévu, l'expression de la variance de f_v est complexe. Cependant, il est à remarquer que de nombreux termes se répètent et que tous se calculent de façon simple et routinière, sauf le terme $\sum_{P_v} \Pr(p|j)\, D(p,j)$ qui exprime la corrélation entre les différents arcs du réseau. Aussi, nous allons réserver l'Appendice [illegible] à l'élaboration d'une technique pour l'évaluation de ce terme.

3.8 Applications

Le champ d'applications de ce modèle sera délimité par les développements à venir. Il nous est cependant déjà possible d'énumérer quelques-unes des sphères d'applications.

Il faut différencier deux champs d'applications bien distincts: celui où le modèle se situe au centre d'un processus itératif comme le schématise la figure suivante et celui où on considère le modèle complet en soi, ce qui correspond à une seule itération du processus général.



L'application la plus immédiate se situe dans le contexte itératif. Nous avons alors un modèle d'affectation dynamique dépendant du niveau de volume. La structure dynamique du flot ainsi que la dépendance du volume sur l'affectation du trafic permettent l'étude de la formation et de l'évolution de la congestion sur les éléments du réseau; elles donnent en outre une image plus détaillée et plus exacte de la réalité. Malheureusement, le coût qu'implique une telle étude est énorme.

Si on s'attarde à une seule itération où on exige que les fonctions sur les arcs ne dépendent pas du volume qu'on affecte, on note les appli-

cations qui suivent.

En premier lieu, ce modèle dynamique peut souvent s'appliquer sans trop d'efforts là où les modèles d'affectation du type "tout ou rien" ou du type probabiliste stationnaire sont employés, permettant ainsi une représentation plus fidèle de la réalité.

Ce modèle permet également, en une seule itération, l'affectation dynamique suivant le volume de trafic dans les cas où on veut affecter un surplus de voyageurs sur un réseau déjà achalandé. Ce surplus doit être tel qu'il ne change pas de façon sensible le flot de trafic présent sur le réseau ou qu'il est composé de voyageurs ne pouvant pas d'adapter aux nouvelles conditions de circulation engendrées par une situation inhabituelle. Cette situation se présente, par exemple, à celui qui veut connaître sur un réseau la répartition des employés quittant un immeuble pour retourner à leur domicile, leur nombre étant relativement petit. Le second cas concerne celui qui veut connaître le point de congestion créé par les gens qui entrent chez eux à la fin d'un spectacle: ces derniers, ignorant l'état du réseau sous ces nouvelles conditions, perçoivent le réseau comme il est habituellement à ces heures.

Comme l'a noté Florian (1973), ce modèle sert aussi pour l'affectation des utilisateurs d'un réseau de transport en commun, car les fonctions sur les sections des différentes lignes varient selon les heures de la journée mais ne dépendent pas des utilisateurs sur ces lignes. La congestion est présente mais indépendante des usagers qui doivent utiliser le réseau de transport en commun.

3.9 Conclusion

Il importe de raffiner les modèles utilisés présentement et qui représentent supposément la réalité. Le but de ce chapitre est de développer un outil adéquat qui amènera les gens faisant de la planification en transport à mieux comprendre le phénomène du transport et à trouver des solutions adaptées à la réalité.

Nous avons éliminé une hypothèse irréaliste acceptée par la plupart des modèles utilisés aujourd'hui, à savoir l'hypothèse de l'état stationnaire du flot de trafic. De plus, dans un processus itératif, ce modèle résoud le problème de l'affectation dynamique dépendant du volume de trafic. Il reste maintenant à puiser dans cette théorie exposée et à la confronter à la réalité pour apprécier la portée de la méthode présentée.

L'utilisation de ce modèle demande un prélèvement assez important de données et la mise en place de nombreux programmes d'ordinateur; aussi, l'aspect expérimental fera l'objet d'études futures et c'est à ce moment que les conclusions seront véritablement émises.

CHAPITRE IV

L'EVALUATION DE LA MATRICE ORIGINE-DESTINATION ET DE CERTAINS ELEMENTS DU RESEAU

4.1 Historique

Jusqu'à ce jour, les volumes de trafic observés sur les tronçons de chemins ont servi presque exclusivement à étudier le contrôle de la circulation et de la congestion sur les routes et à alimenter divers programmes de simulation dont DINET et UTCS . Ce chapitre propose une méthodologie permettant d'utiliser ces données en vue de la planification en transport, là où l'enquête origine-destination s'impose habituellement. Nous montrons que la charge observée sur les tronçons de routes contient suffisamment d'informations pour s'acquitter de cette tâche et aussi pour préciser certains attributs du réseau.

4.2 Affectation des voyages

Un système affectant des voyages se compose d'un algorithme qui, appliqué à un bloc de données appelé données d'entrée, produit une répartition de trafic. La figure 4.1 illustre le schéma d'un tel système.

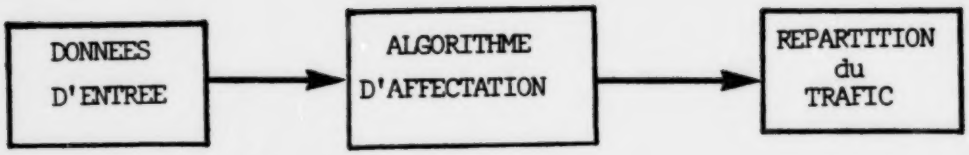


figure 4.1 Un système d'affectation des voyages.

Les données d'entrée requises par les différents algorithmes d'affectation varient peu d'un système à l'autre. De plus, il est possible de les répartir en trois catégories, soit les données décrivant le réseau, les données se rapportant aux attributs du réseau et enfin, la matrice origine-destination qui comporte l'information venant de la distribution des voyages.

On appelle réseau de transport la représentation d'un ensemble de routes, de boulevards, d'intersections, etc. Sa description fait appel au langage de la théorie des graphes. Potts et Oliver (1972) traitent abondamment de ce sujet au chapitre II de leur volume. La première catégorie de données d'entrée décrivant le réseau fait donc appel à des notions bien connues.

La deuxième rassemble des données telles que la longueur des arcs, leur capacité, leur temps de parcours. Elle fournit des informations indispensables quant à l'utilité relative des éléments du réseau et sert à reproduire de façon strictement mathématique le comportement des usagers sur le réseau. Cette catégorie fait appel au langage de la théorie des flots. La détermination et l'évaluation des éléments quantitatifs du réseau, comme le souligne le Highway Research Board (1965) dans son rapport

85, sont des étapes importantes de la planification en transport.

La demande en voyages entre chaque paire origine-destination constitue la dernière catégorie de données d'entrée accédant à l'algorithme d'affectation. L'évaluation des demandes résulte d'une analyse séparée, soit celle de la distribution des voyages, et constitue une des étapes primordiales du processus de planification. Un manque de précision à cette étape rend non valides tous les résultats subséquents quelle que soit la qualité de l'algorithme d'affectation utilisé. Plusieurs méthodes sont disponibles pour évaluer la matrice origine-destination et elles sont rapportées par Wilson (1970) et par Potts et Oliver (1972) ainsi que par les auteurs dont ils font leurs références.

Un algorithme d'affectation est une procédure qui estime pour chaque arc du réseau le nombre de voyages empruntant cet arc. Les systèmes d'affectation diffèrent les uns des autres surtout par leur algorithme d'affectation respectif.

Pour des données d'entrée précisées, nous dirons qu'un système d'affectation est fiable et réaliste s'il produit une répartition du trafic s'accordant aux volumes de trafic observés sur les arcs. Si nous adoptons une mesure de concordance ou de similitude exprimant la ressemblance entre la répartition de trafic issue de l'algorithme et celle observée, nous pourrons juger de la qualité de divers algorithmes d'affectation . D'autre part, cette mesure de concordance permettra de comparer la pertinence de plusieurs matrices origine-destination pour un même algorithme d'affectation; mieux encore, il nous sera possible de choisir la matrice origine-destination préférable du point de vue de la mesure de concordance adoptée, c'est-à-dire la matrice origine-destination qui

minimise la mesure de concordance.

4.3 Mesure de concordance

Au préalable, nous devons définir certains éléments de notation. Nous dénotons par O-D l'ensemble des paires ordonnées de sommets dans V qui sont des origines et/ou des destinations. q , élément de O-D, est égale à (O_q, D_q) où O_q est une origine et D_q une destination. Y_q représente le nombre de voyages commençant à l'origine O_q et se terminant à la destination D_q . L'ensemble de tous les Y_q est dénoté par Y . Nous appelons β l'ensemble des éléments quantitatifs associés au réseau.

Un algorithme d'affectation A estime pour chaque arc e une charge X_e . L'ensemble de X_e , noté X , forme la répartition du trafic. X_e est fonction du réseau G , de l'ensemble des attributs du réseau β , de l'ensemble des demandes en voyages entre les paires origine-destination Y ainsi que de l'algorithme d'affectation A . Cette dépendance s'exprime comme suit:

$$X_e = H_A[G, \beta, Y, e] \quad .$$

Il faut remarquer que G, β de même que Y représentent respectivement les trois catégories de données d'entrée.

Nous dénotons par x_e la charge observée sur l'arc e et par E' l'ensemble des arcs e pour lesquels x_e est disponible. E' est un sous-ensemble de E . Nous n'exigeons pas qu'il y ait conservation de flot pour la répartition partielle qu'engendrent les x_e . Une mesure

de concordance C entre la répartition estimée $X = \{X_e\}$ et la répartition partielle observée $x = \{x_e\}$ est une fonction non négative dotée des propriétés suivantes:

$$C(x,X) = 0 \quad \text{si} \quad x_e = X_e \quad \text{pour tous les} \quad e \quad \text{dans} \quad E', \tag{4.1}$$

$$C(x,X) = C(X,x) \tag{4.2}$$

et la fonction C doit croître lorsque l'écart entre x et X s'accentue. Ces vagues restrictions laissent une grande latitude quant au choix de C . Nous suggérons, dans le cadre de ce travail, la définition:

$$C(x,X) = \sum_{E'} (x_e - X_e)^2 . \tag{4.3}$$

Evidemment, d'autres définitions sont possibles et ont été effectivement utilisées dans des contextes similaires. Nous trouvons dans le livre de Jardine et Sibson (1971) une discussion complète de ces mesures ainsi que plusieurs références concernant ce sujet.

La mesure C est la somme des carrés des différences entre la charge observée et la charge estimée pour les arcs e dans E' et elle est fonction de A, G, β, Y et E' . Nous convenons que l'algorithme A est préférable à l'algorithme A' lorsque la valeur de C est plus petite pour A que pour A' , toutes choses étant égales par ailleurs.

Pour A, G, β et E' donnés, une estimation de la vraie matrice origine-destination est l'ensemble $\hat{Y}$ qui minimise la mesure de concordance définie à l'équation (4.3). Cette estimation non linéaire des moindres carrés pour la matrice origine-destination peut être obtenue, du moins en théorie, pour tous les systèmes d'affectation; cette méthode

générale convient également pour estimer certains éléments de β . Nous pouvons nous référer au volume de Draper et Smith (1967) pour obtenir des précisions quant aux modèles de régression non linéaires impliqués par ces estimations.

Dans un contexte général, cette méthode est peu efficace à cause de sa complexité et du très grand nombre d'opérations qu'elle nécessite. En effet, certains algorithmes d'affectation sont très onéreux en termes de temps machine pour réaliser une répartition du trafic et ceci, même sur des réseaux restreints. Les difficultés engendrées par cette méthode appartiennent beaucoup plus au domaine opérationnel qu'au domaine conceptuel. Afin de contourner ces difficultés, nous allons nous confiner aux systèmes d'affectation mettant en cause des modèles d'affectation proportionnels.

4.4 Modèles d'affectation proportionnels

Pour un modèle d'affectation, un graphe G et un ensemble β donnés, nous dénotons par $X_e^q(Y_q)$ la charge affectée sur l'arc e pour une matrice origine-destination dont tous les éléments sont nuls, mis à part l'élément correspondant à la paire q qui est égale à Y_q . Un modèle d'affectation est dit proportionnel lorsqu'il possède les propriétés suivantes pour tout G, β et Y :

$$X_e = \sum_{O\text{-}D} X_e^q(Y_q) \quad \text{pour tout } e \text{ dans } E \tag{4.4}$$

et

$$X_e^q(Y_q) = Y_q \, X_e^q(1) \quad \text{pour tout } e \text{ dans } E . \tag{4.5}$$

La propriété (4.4) signifie que le flot total affecté à chacun des arcs du réseau doit égaler la somme des flots affectés à chacun des arcs lorsque le modèle traite séparément et successivement une seule paire origine-destination. Le mot "proportionnel" attribué à cette classe de modèles découle de la propriété (4.5). Cette dernière, en accord avec la propriété (4.4), exprime entre autres choses que si tous les éléments de la matrice origine-destination sont doublés, la charge sur les arcs du réseau sera par conséquent le double de celle obtenue avec la matrice initiale. Ces deux propriétés sont communes à tous les modèles d'affectation ne considérant pas de façon explicite ou implicite des contraintes de capacité sur les arcs, soit les modèles d'affectation sans capacité. Parmi ces modèles, nous retrouvons le modèle du "tout ou rien", les modèles de dispersion ainsi que le modèle probabiliste proposé par Dial.

Si nous utilisons les propriétés (4.4) et (4.5), l'estimation de la matrice origine-destination se ramène à déterminer l'ensemble des Y_q minimisant la fonction:

$$\sum_{E'} \left[x_e - \sum_{O\text{-}D} Y_q \, X_e^q (1) \right]^2 \quad .$$

Les $\hat{Y}_q$ sont les estimateurs des moindres carrés du modèle de régression:

$$x_e = \sum_{O\text{-}D} Y_q \, X_e^q (1) + \varepsilon_e \quad \text{pour tout} \quad e \quad \text{dans} \quad E' \quad . \qquad (4.6)$$

La répartition du trafic $\hat{X}$ définie:

$$\hat{X} = \{ \hat{X}_e \mid \hat{X}_e = \sum_{O\text{-}D} \hat{Y}_q X_e^q (1) \quad \text{pour} \quad e \quad \text{dans} \quad E \}$$

est celle qui, parmi l'ensemble des répartitions de trafic possibles avec le modèle utilisé, approche le mieux la répartition partielle observée, suivant la mesure des moindres carrés. La répartition $\hat{X}$ satisfait les équations de conservation du flot tandis que $\hat{Y}$ est la matrice origine-destination préférée dans ce contexte.

De plus, si les variables aléatoires ε_e sont indépendantes et identiquement distribuées suivant une normale $N(0,\sigma^2)$, nous avons alors un problème classique de régression linéaire et la distribution de $\hat{Y}_q$ est connue pour tout q dans O-D .

Il est à noter que pour le modèle du "tout ou rien", $X_e^q(1)$ prend la valeur 1 ou 0 selon que l'arc e appartient ou non au plus court chemin entre la paire origine-destination q . Pour le modèle probabiliste étudié précédemment, $X_e^q(1)$ est la probabilité qu'un usager voyageant entre O_q et D_q utilise l'arc e .

4.5 Modèle de gravité

Un inconvénient de la méthode suggérée est la grande taille de l'échantillon requis. En effet, le nombre de paires origine-destination égale souvent le produit du nombre d'origines et du nombre de destinations. Il en résulte que le nombre de paramètres à estimer et conséquemment la taille de l'échantillon nécessaire peuvent croître très rapidement. L'estimation devient alors fort coûteuse en termes de temps machine.

Nous pouvons éviter cette difficulté de multiples façons, la plus triviale étant de réduire le nombre d'origines et le nombre de destina-

tions en les regroupant. Une telle simplification engendre cependant une perte de précision du modèle.

Une autre solution consiste à exprimer la matrice origine-destination en fonction de paramètres associés aux origines et aux destinations. Le nombre de paramètres à estimer est alors proportionnel à la somme du nombre d'origines et du nombre de destinations. Par exemple, Y_q la demande en voyages pour la paire origine-destination q peut être formulée ainsi:

$$Y_q = a_{O_q} \, b_{D_q} \, g(u, C_q) \qquad (4.7)$$

où

u est un paramètre inconnu qui caractérise la fonction g ;

C_q est une quantité connue exprimant le coût relatif pour passer de O_q à D_q ; elle est une pondération indiquant la proximité relative de chacune des paires origine-destination. Plusieurs formules ont été suggérées pour calculer C_q ;

$g(u,C_q)$ est une fonction connue de u et de C_q . Les diverses façons de définir la fonction g engendrent différents modèles de distribution. Hyman (1969) discute de ces modèles;

a_{O_q} est un paramètre inconnu associé à l'origine O_q ;

b_{D_q} est un paramètre inconnu associé à la destination D_q .

La formule (4.7) est analogue à celle utilisée par le modèle de gravité. Wilson (1970) discute de ce modèle. Le modèle de régression (4.6) peut alors s'écrire:

$$x_e = \sum_{O\text{-}D} a_{O_q} b_{D_q} g(u,C_q) X_e^q(1) + \varepsilon_e \quad \text{pour tout } e \text{ dans } E$$

avec la restriction que les paramètres a et b doivent être positifs. L'estimation des différents paramètres par la mesure des moindres carrés se ramène à la minimisation de la fonction:

$$\sum_{E'} \left[x_e - \sum_{O\text{-}D} a_{O_q} b_{D_q} g(u,C_q) X_e^q(1) \right]^2 . \qquad (4.8)$$

Puisque les x_e sont positifs, la contrainte de positivité sur les a et les b est implicitement satisfaite. Le problème d'estimation est essentiellement un problème de régression non linéaire. Il faut remarquer que les estimations des a et des b ne sont pas uniques; nous pouvons multiplier les a par une constante et diviser les b par la même constante sans changer la valeur de la fonction (4.8). Aussi, il faut être attentif lors de l'estimation de ces paramètres.

Dans une publication datée de 1971, Lawton et Sylvestre montrent comment minimiser une fonction du type (4.8) de façon efficace: ils concluent que pour des b et un u donnés, le problème de minimiser (4.8) revient à un problème de régression linéaire par rapport aux a . Ils suggèrent alors de remplacer les a par la solution de la régression linéaire s'exprimant en fonction des b et de u , et de minimiser (4.8) par rapport aux b et à u seulement. Cette procédure très simple réduit

le nombre de paramètres à estimer de façon non linéaire et accélère la convergence de la méthode.

Si nous appelons respectivement $\hat{a}$, $\hat{b}$ et $\hat{u}$ les estimations de a , b et u , l'estimation de Y_q peut s'écrire:

$$\hat{Y}_q = \hat{a}_{O_q} \hat{b}_{D_q} g(\hat{u}, C_q)$$

et le volume estimé sur l'arc e est donné par:

$$\hat{X}_e = \sum_{O\text{-}D} \hat{Y}_q X_e^q \quad (1) \quad .$$

4.6 Modèle de régression linéaire multiple

Une troisième façon de réduire le nombre de paramètres à estimer dans la formule (4.6) est d'exprimer la matrice origine-destination suivant un modèle de régression linéaire multiple. Comme le propose Osofsky (1959), la matrice origine-destination peut être formulée comme une combinaison linéaire de facteurs socio-économiques:

$$Y_q = a_o + \sum_{i=1}^{m} a_i z_{iq} \quad \text{pour } q \text{ dans } O\text{-}D \qquad (4.9)$$

où

z_{iq} est la valeur de la $i^{ème}$ variable socio-économique pour la paire origine-destination;

a_i est un paramètre inconnu associé à la $i^{ème}$ variable socio-économique;

a_o est une constante inconnue représentant la moyenne globale.

Les variables z_{iq} représentent des données socio-économiques du genre: nombre de personnes possédant une automobile dans la zone contenant le sommet O_q, nombre d'unités de logement dans la zone contenant D_q, index de l'activité économique dans la zone contenant D_q ; ces données socio-économiques sont pondérées par un facteur qui est fonction de la distance entre les sommets O_q et D_q.

Si nous substituons Y_q dans le modèle de régression (4.6) par son équivalent dans la formule (4.9), nous pouvons écrire:

$$x_e = \sum_{O\text{-}D} \left(a_o + \sum_{i=1}^{m} a_i z_{iq}\right) X_e^q(1) + \varepsilon_e \quad \text{pour } e \text{ dans } E'$$

ou encore

$$x_e = a'_o + \sum_{i=1}^{m} a_i \left[\sum_{O\text{-}D} z_{iq} X_e^q(1)\right] + \varepsilon_e \quad \text{pour } e \text{ dans } E' .$$

Le problème de l'estimation des a se ramène à un problème de régression linéaire avec peu de paramètres. Comme l'a noté Osofsky (1959), le désavantage de ce modèle est qu'il peut engendrer des estimations $\bar{X}_e$ négatives.

4.7 Modèles d'affectation non proportionnels

La méthodologie proposée à la section 4.3 s'applique sans aucune difficulté théorique aux modèles d'affectation non proportionnels: ces modèles, dont celui de Dafermos (1969), considèrent de façon implicite ou explicite des contraintes de capacités sur les arcs.

Empruntant la notation de la section 4.3, l'estimation de la matrice origine-destination consiste à minimiser, par rapport à Y, la somme des carrés des différences entre la charge observée et la charge estimée sur divers arcs du réseau, c'est-à-dire qu'il faut résoudre:

$$\min_{Y} \left[\sum_{E'} (x_e - X_e)^2 \right] = \min_{Y} C(x,X) \quad . \tag{4.10}$$

Pour un système d'affectation donné, X est la fonction de Y. La relation entre X et Y est de type algorithmique. A cause de la complexité des algorithmes, cette relation ne peut pas être explicitée en règle générale. Il est toutefois possible de résoudre directement ce problème en utilisant la méthode proposée par Powell (1964) qui trouve le minimum d'une fonction à plusieurs variables sans calculer les dérivées. Il suffit d'évaluer $C(x,X)$ pour différentes valeurs de Y, ce qui est réalisé à l'aide de l'algorithme d'affectation.

Cette méthode générale de résolution n'est pas efficace, car elle ne tient pas compte de la structure du problème en cause. Il reste beaucoup à faire dans ce domaine de la recherche.

4.8 Généralisation et conclusion

Nous nous rendons vite compte qu'il est possible d'adapter cette méthodologie à bien d'autres problèmes dont l'estimation des éléments quantitatifs du réseau (les éléments de β) et la calibration du modèle d'affectation lui-même. L'ajustement des fonctions de coûts utilisées

dans les modèles d'affectation à plusieurs commodités avec coûts convexes ainsi que l'évaluation de l'importance relative des diverses composantes entrant dans les fonctions de coûts constituent d'autres exemples.

Pour simplifier la présentation, nous nous sommes restreints à un seul mode de transport et à une seule catégorie d'usagers. Evidemment, la méthode élaborée dans ce chapitre se généralise sans difficulté pour permettre l'étude de situations plus complexes.

A cause de la complexité et de l'aspect aléatoire des phénomènes inhérents aux problèmes de transport, l'analyse en transport ne peut se faire adéquatement sans l'aide de la technique statistique. Cette dernière analyse consiste à identifier et à séparer les composantes aléatoires des composantes déterministes ainsi qu'à mesurer la fiabilité du modèle suggéré. La méthode proposée permet une telle analyse.

Les avantages de l'utilisation des volumes observés sur les arcs du réseau se situent sur deux plans: d'un point de vue économique, ces données sont accessibles à un faible coût et d'un point de vue conceptuel, elles offrent, entre autres, la possibilité d'unifier dans une seule analyse la génération, la distribution et l'affectation des voyages.

CHAPITRE V

L'ESTIMATION DE LA DISTRIBUTION DE LA DUREE D'UN PROGRAMME

5.1 Représentation d'un programme par un graphe

Nous appelons programme un ensemble d'activités concourant à la réalisation d'un objectif. Les durées des activités, qui peuvent être déterminées ou aléatoires, sont supposées connues de même que les relations d'ordre les concernant. Une activité consume des ressources telles que le temps, occupe une position bien définie à l'intérieur d'un programme et est tout à fait délimitée par un point initial et un point terminal. Les points initiaux et terminaux des activités sont appelés événements. En principe, un événement est un point instantané dans le temps.

Un réseau PERT, G , est une représentation graphique d'un programme: à chaque activité du programme est substitué sur G un arc auquel on associe une variable positive; cette dernière indique le temps nécessaire pour accomplir l'activité représentée par l'arc. Le graphe est défini sans échelle de grandeur et par conséquent, aucune signification n'est attribuée à la longueur des arcs. Un arc sur G ne sert qu'à illustrer l'ordre chronologique des activités. Le sommet initial d'un arc correspond au déclenchement de l'activité illustrée par l'arc et le sommet terminal, à son achèvement. Les sommets d'un réseau PERT sont les événements. La position d'un arc sur un réseau

PERT est dictée par la règle suivante: toutes les activités associées aux arcs entrant dans un sommet doivent être complètement exécutées avant que celles associées aux arcs sortant de ce même sommet puissent débuter. Il s'ensuit que G est un graphe orienté sans cycle, sans quoi une activité pourrait se faire suite à elle-même.

Soit V l'ensemble des sommets et E l'ensemble des arcs sur G . Nous définissons la fonction n qui fait correspondre à chaque arc e dans E une paire ordonnée de sommets distincts dans V , soit (v_i,v_j). n induit de façon naturelle les fonctions n_1 et n_2 définies comme suit:

$$n(e) = (v_i,v_j) = [n_1(e), n_2(e)] \quad \text{pour } e \text{ dans } E$$

où v_i et v_j sont respectivement le sommet initial et le sommet terminal de l'arc e . Un arc est indifféremment référé par la lettre e ou par la paire ordonnée de sommets qui lui est associée.

5.2 Approche probabiliste pour l'étude des réseaux PERT

L'ordonnancement d'un programme requiert la prédiction de la durée des activités. Dans un programme, il y a essentiellement deux types d'activités qui illustrent des situations extrêmes; les autres activités ont rapport à des situations intermédiaires. Le premier type d'activités regroupe celles dont il est possible de prévoir avec précision la durée. Les variables attachées aux arcs associés à ces activités sont des nombres uniques connus et incontestés ou des estimations uniques des temps moyens requis pour réaliser les activités en question. Ce type concerne

tout particulièrement la méthode CPM et, à cet effet, il nous est possible de nous référer à un des nombreux livres ou articles traitant de ce sujet. Battersby (1967), Kaufmann et Desbazeille (1969) ainsi que Moder et Phillips (1970) ont travaillé dans ce sens.

D'autre part, on trouve les activités qui n'ont jamais, ou presque, été réalisées et/ou qui comportent un fort coefficient d'incertitude. Les programmes de développement et de recherche sont composés d'activités appartenant à ce type. Il est alors pertinent de considérer les variables associées aux durées de ces activités comme aléatoires. Ce type d'activités se rapporte plus précisément à la méthode PERT. Au cours de cette étude des réseaux PERT, nous supposons que toutes les activités considérées appartiennent à ce type probabiliste. Comme l'ont noté Moder et Philipps (1970), quelques-uns des avantages à introduire une approche probabiliste pour l'étude des réseaux PERT sont l'opportunité d'inclure dans l'organisation d'un programme les variations dues aux incertitudes et le calcul de la probabilité qu'un programme soit terminé avant une date limite.

Nous affectons à chaque arc e dans E une variable aléatoire t_e qui indique le temps de réalisation de l'activité associée à cet arc e. f_e, F_e et Ψ_e sont respectivement la fonction de densité, la fonction de distribution et la fonction caractéristique de la variable aléatoire t_e. Nous supposons que les t_e sont indépendantes.

Dans la première partie de ce chapitre, nous assumons, sans pour autant nous éloigner des généralités, que le réseau PERT G a un seul sommet-source S dans V qui correspond au commencement du programme

ainsi qu'un seul sommet-objectif F dans V qui correspond à la réalisation du programme. Tout réseau PERT peut satisfaire à cette hypothèse en ajoutant, s'il y a lieu, des arcs fictifs dont les variables sont des constantes égales à zéro.

Soit P_v l'ensemble des r_v chemins p entre S et v dans V . Pour faciliter la lecture, nous allons remplacer:

P_F par P ;

$\sum_{e \text{ dans } E}$ par $\sum_{E}$;

$\sum_{p \text{ dans } P_v}$ par $\sum_{P_v}$

et

$\prod_{e \text{ dans } E}$ par $\prod_{E}$.

Nous définissons la durée minimale du chemin p dans P_v par:

$$t_p = \sum_E \delta_{ep}\, t_e \quad \text{pour } p \text{ dans } P_v$$

avec

$$\delta_{ep} = \begin{cases} 1 & \text{si e appartient à p} \\ 0 & \text{autrement.} \end{cases}$$

t_p est une variable aléatoire ayant f_p comme fonction de densité, F_p comme fonction de distribution et ψ_p comme fonction caractéristique. Soit T le temps nécessaire pour atteindre le sommet F , c'est-à-dire que toutes les activités associées aux arcs entrant dans le sommet F sont réalisées. T , la durée du programme, est la plus longue des durées minimales des chemins p dans P :

$$T = \max_{p \text{ dans } P} t_p .$$

f est la fonction de densité de T, F sa fonction de distribution et ψ sa fonction caractéristique. Un problème important est l'évaluation de $F(x)$, la fonction de distribution de la durée du programme. Nous avons:

$$F(x) = Pr[T \leq x] = Pr[\max_{p \text{ dans } P} t_p \leq x] = Pr[t_p \leq x \text{ pour tout } p \text{ dans } P] .$$

5.3 Sommaire de deux approches existantes pour estimer la distribution de la durée d'un programme

On reconnaît deux approches de base pour estimer la fonction de distribution de la durée d'un programme dans la théorie des réseaux PERT: l'approche de Moder et Phillips * et celle de Martin.

L'approche de Moder et Phillips utilise le théorème central limite à savoir que T comme somme de variables aléatoires a une distribution approximativement normale. Les paramètres de la distribution normale associés à T dépendent de la durée des activités appartenant au chemin critique tel que défini dans la méthode CPM où t_e est remplacée par $E[t_e]$, la moyenne de f_e. Nous supposons en outre que le nombre d'activités composant le chemin critique est suffisamment grand et que la durée moyenne du chemin critique s'écarte considérablement de celle des autres chemins dans P_F.

---

* La première approche présentée est appelée "approche de Moder et Phillips", car ces derniers l'ont bien résumée. Cependant, elle fut développée par Malcolm, Roseboom, Clark et Fazar (1959), Clark (1961) ainsi que par Van Slyke (1963) pour n'en nommer que quelques-uns.

L'inconvénient majeur de cette approche réside dans le fait qu'elle considère un seul chemin critique. Généralement, plusieurs chemins peuvent être critiques pour diverses réalisations des variables aléatoires associées aux durées des activités. Il en résulte que cette approche entraîne, pour la vraie fonction de distribution de la durée du programme, une estimation qui est toujours biaisée de façon optimiste. D'un point de vue pratique, un biais pessimiste serait préférable. De plus, le biais de cette estimation, non borné, dépend de la structure du réseau PERT impliqué: il croît avec le nombre de chemins parallèles et lorsque la durée moyenne des chemins parallèles se rapproche de celle du chemin critique. MacCrimmon et Ryavec (1964) ainsi que Moder et Phillips (1970) ont discuté de ces inconvénients.

Les désavantages de l'approche de Moder et Phillips qui considère un seul chemin critique peuvent être réduits grâce aux techniques de simulation. Ces améliorations sont cependant dispendieuses en termes de temps machine. Une approche éliminant les désavantages précédents, n'utilisant pas les techniques de simulation et produisant une meilleure estimation avec des bornes à cette même estimation serait bienvenue.

Martin (1965) a présenté une méthode pour évaluer la fonction de densité de la durée d'un programme dans la théorie des réseaux PERT lorsque les fonctions de densité associées aux arcs du réseau PERT peuvent être exprimées sous forme polynômiale. Cette méthode se compose de deux étapes: la première permet de transformer tout réseau orienté sans cycle en un réseau série-parallèle défini par Martin lui-même. Cette transformation s'accomplit en décrivant un réseau PERT à l'aide

d'un arbre où certains sommets et certains arcs sont reproduits plusieurs fois s'il y a lieu. La seconde étape met en cause un algorithme qui réduit systématiquement un réseau série-parallèle à un simple arc possédant une fonction de densité qui caractérise la durée du programme original.

L'intérêt de l'approche de Martin réside dans l'exactitude des résultats. En effet, le seul biais provient de l'approximation des fonctions de densité associées aux durées des activités par des polynômes. Malheureusement, cette précision exige des calculs considérables ainsi que des développements complexes même pour des réseaux PERT très simples. L'exemple que Martin présente illustre bien ces inconvénients.

Nous avons résumé brièvement les deux tendances extrêmes les plus populaires pour l'estimation de F. Il faut noter qu'il existe d'autres approches fondées sur les idées inhérentes à ces deux tendances. Charnes (1963), Hartley et Wortham (1966) ainsi que Ringer (1969) ont développé quelques-unes d'entre elles.

5.4 Borne inférieure pour la fonction de distribution de la durée d'un programme

Le but de cette section est de présenter une nouvelle approche se situant entre les deux précédentes: elle permet d'obtenir une meilleure approximation que celle obtenue par l'approche de Moder et Phillips; elle ne comporte pas les inconvénients des techniques de simulation et elle est aussi plus économique que la méthode de Martin.

Nous avons déjà obtenu:

$$F(x) = Pr[t_p \leq x \quad \text{pour tout} \quad p \quad \text{dans} \quad P] \quad .$$

Or, l'inégalité de Bonferroni, Feller (1968)*, nous permet d'écrire:

$$\lambda(x) = \max\left\{0,\ 1 - \sum_P Pr[t_p > x]\right\} \leq F(x) \leq$$

$$\min\left\{1,\ 1 - \frac{3}{2}\sum_P Pr[t_p > x] + \sum_P \sum_{P'} \frac{Pr[t_p > x \text{ et } t_{p'} > x]}{2}\right\}$$

où $P' = P$.

Nous allons nous attarder à la borne inférieure qui peut se formuler:

$$\lambda(x) = \max\left\{0,\ 1 - \sum_P [1 - F_p(x)]\right\} \leq F(x) \quad .$$

Si $r\ (= r_P)$ est le nombre de chemins dans P, nous avons:

$$\lambda(x) = \max\left\{0,\ 1 - r + r \sum_P \frac{F_p(x)}{r}\right\} \quad .$$

Empruntant le lemme de Feller (1971)**, nous obtenons que $\sum_P \frac{F_p(x)}{r}$ est une fonction de distribution dont la fonction caractéristique est $\sum_P \frac{\psi_p(Z)}{r}$. Si nous définissons $\prod_x^{-1}[\gamma]$ comme la fonction caractéristique inverse de γ évaluée en x, nous arrivons à:

$$\lambda(x) = \max\left\{0,\ 1 - r + r \prod_x^{-1}\left[\sum_P \frac{\psi_p(Z)}{r}\right]\right\} .$$

*Feller, William, An Introduction to Probability Theory and its Applications, Vol. 1, troisième édition, John Wiley & Sons, Inc., p. 110, 1968.

** Idem, Vol. II, deuxième édition, p. 504, 1971.

L'indépendance des variables aléatoires t_e permet d'écrire:

$$\lambda(x) = \max\left\{0,\ 1 - r + r \prod_{x}^{-1}\left[\frac{\phi(Z)}{r}\right]\right\}$$

où $\phi_v(Z) = \sum_{P_v} \prod_{E} \psi_e(Z)^{\delta_{ep}}$ pour v dans V, avec la convention que

$\phi_S(Z) = 1$ et que $\phi_F = \phi$. Il faut remarquer que les fonctions ϕ_v sont analogues aux fonctions w rencontrées aux chapitres II et III.

La fonction $\phi(Z)$ doit être évaluée pour plusieurs valeurs de Z. Nous allons maintenant développer un algorithme qui permet d'évaluer la fonction $\phi(Z)$ pour différentes valeurs de Z d'une façon très simple en balayant une seule fois le réseau PERT.

Sachant qu'un réseau PERT ne contient pas de cycle et partant de la définition de $\psi_e(Z)$, nous pouvons en déduire:

$$\phi_v(Z) = \sum_{E_v} \frac{\phi(Z)}{n_1(e)} \psi_e(Z) \qquad \text{pour tout } v \text{ dans } V$$

où $E_v = \{ e \text{ dans } E \mid n_2(e) = v \}$.

Cette formule nous autorise à calculer les quantités $\phi(\cdot)$ de façon séquentielle. Les $\phi_v(\cdot)$ sont évaluées en parcourant les sommets du réseau PERT dans un ordre topologique de telle sorte que le sommet v_i est parcouru avant le sommet v_j si (v_i,v_j) appartient à E. L'algorithme progresse en calculant les $\phi_v(\cdot)$ à chaque sommet v rencontré

et prend fin lorsque le sommet F est atteint. Il suffit de balayer une seule fois le réseau PERT pour obtenir les $\phi(\cdot)$.

Il est à noter que si nous voulons estimer la fonction de distribution associée au temps requis pour accomplir partiellement un programme, ou que si le réseau PERT a plusieurs objectifs finals, il faut arrêter l'algorithme quand tous les sommets pertinents ont été balayés. Ainsi, en balayant une seule fois le réseau PERT, il est possible d'obtenir sans trop d'effort une borne inférieure pour la fonction de distribution du temps nécessaire pour atteindre un des sommets du réseau PERT.

La fonction caractéristique inverse de $\frac{\phi(Z)}{r}$ peut être calculée de façon efficace en utilisant l'approximation proposée par Davies (1973). L'approximation de Davies remplace la fonction $\int_x^{-1}$ de la façon suivante:

$$\lambda(x) = \max\left\{0,\ 1 - \frac{r}{2} - \sum_{k=0}^{K} \frac{IM\left[\phi((k+0.5)\,\Delta)\, e^{-i(k+0.5)\,\Delta\, x}\right]}{\pi(k+0.5)}\right\} < F(x)$$

où la fonction IM[NC] rend la partie imaginaire du nombre complexe NC . Le choix de K et de Δ est fait de telle sorte que:

$$\max\left\{Pr[T < x - \tfrac{2\pi}{\Delta}]\ ,\ Pr[T > x + \tfrac{2\pi}{\Delta}]\right\}$$

et que l'erreur par troncation soient tous les deux inférieurs à la moitié de l'erreur maximale acceptée dans l'approximation de $\int_x^{-1}$. On doit noter que la fonction $\phi(\cdot)$ ne dépend pas de x . Ces précisions complètent le développement théorique qui produit une borne inférieure

λ pour F .

Il est également possible d'obtenir des bornes pour les différents moments de F : nous allons dériver explicitement la moyenne et la variance de F .

Rao (1971) soutient que si T est une variable aléatoire non négative et que si $E[T]$, la moyenne de F , existe,

$$E[T] = \int_0^\infty [1 - F(x)]\, dx \ .$$

En nous servant de la borne $\lambda(x) \leq F(x)$, nous obtenons:

$$\overline{E[T]} = \int_0^\infty [1 - \lambda(x)]\, dx \geq E[T]$$

qui est une borne supérieure pour la moyenne du temps requis pour réaliser le programme. Une borne inférieure $\underline{E[T]}$ pour $E[T]$ est obtenue par la méthode CPM où $E[t_e]$ se substitue à t_e sur les arcs du réseau PERT. Au chapitre VI, nous démontrerons qu'il existe de meilleurs bornes inférieures que $\underline{E[T]}$. Cependant, afin de simplifier notre exposé, nous réservons l'étude de ces bornes au chapitre suivant.

La formule de la moyenne se généralise pour donner les autres moments:

$$E[T^n] = n \int_0^\infty x^{n-1} [1 - F(x)]\, dx \ .$$

Particulièrement pour le deuxième moment, nous obtenons:

$$\overline{E[T^2]} = 2 \int_0^\infty x[1 - \lambda(x)]\, dx \geq E[T^2] \ .$$

De plus, par définition de T , nous avons:

$$E[T^2] = E\left[\max_{p \text{ dans } P} (t_p^2)\right] .$$

Puisque la fonction $\max_{a \text{ dans } A} (a)$ est convexe et en utilisant l'inégalité de Jensen, nous pouvons écrire:

$$E[T^2] \geq \max_{p \text{ dans } P} E[t_p^2] = \max_{p \text{ dans } P} \left\{ E[t_p]^2 + V[t_p] \right\}$$

où $V[t_p]$ est la variance de t_p .

Si R est le chemin critique obtenu par la méthode CPM, une borne inférieure pour $E[T^2]$ est donnée par:

$$E[T^2] \geq \underline{E[T]^2} + \sum_{E} \delta_{eR} V[t_e] = \underline{E[T^2]} .$$

En usant de la relation $V[T] = E[T^2] - E[T]^2$, nous avons les bornes suivantes pour $V[T]$, la variance de T:

$$\underline{V[T]} = \underline{E[T^2]} - \overline{E[T]}^2 \leq V[T] \leq \overline{E[T^2]} - \underline{E[T]^2} = \overline{V[T]} .$$

5.5 Exemples et commentaires

La méthode décrite à la section précédente a été programmée en FORTRAN EXTENDED sur l'ordinateur CDC CYBER 74 de l'Université de Montréal. L'Appendice III fournit divers renseignements préalables à l'utilisation

du programme REPERT ainsi qu'une description du programme lui-même. On a utilisé REPERT pour étudier le réseau PERT proposé par Martin (1965) et celui proposé par Pritsker et Kiviat (1969).

Martin a étudié le réseau PERT illustré à la figure 5.1.

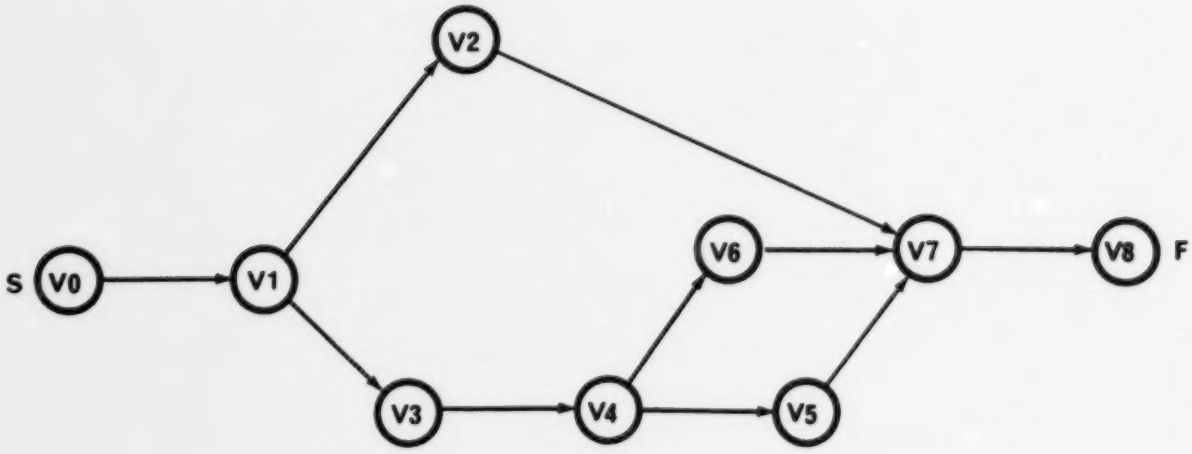


figure 5.1 Exemple d'un réseau PERT proposé par Martin.

Les temps requis pour accomplir les activités suivent une loi de distribution uniforme. Si $f_{ij}(x)$ est la fonction de densité associée à l'arc (v_i,v_j), les données attachées aux arcs du réseau PERT sont les suivantes:

$$f_{01}(x) = 1, \qquad 1 \leq x \leq 2,$$

$$f_{12}(x) = \tfrac{1}{2}, \qquad 2 \leq x \leq 4,$$

$$f_{13}(x) = \tfrac{1}{2}, \qquad 1 \leq x \leq 3,$$

$$f_{27}(x) = \tfrac{1}{4}, \qquad 1 \leq x \leq 5,$$

$$f_{34}(x) = 1, \qquad 2 \leq x \leq 3,$$

$$f_{45}(x) = \tfrac{1}{2}, \qquad 3 \leq x \leq 5,$$

$$f_{46}(x) = \tfrac{1}{4}, \qquad 3 \leq x \leq 7,$$

$$f_{57}(x) = 1, \qquad 1 \leq x \leq 2,$$

$$f_{67}(x) = 1, \qquad 4 \leq x \leq 5,$$

$$f_{78}(x) = \tfrac{1}{2}, \qquad 4 \leq x \leq 6.$$

Nous évaluons la borne $\lambda(x) = \max\left\{ 0, 1 - \sum_{P} \Pr[t_p > x] \right\}$

pour la fonction de distribution de la durée du programme pour diverses valeurs de x. Les résultats sont présentés au tableau 5.1, parallèlement aux valeurs obtenues avec l'approche de Martin.

x \ F(x)	valeurs de Martin	borne inférieure $\lambda(x)$
16	0.00011	0.00000
17	0.00528	0.00000
18	0.04665	0.00000
19	0.17252	0.16111
20	0.38079	0.38003
21	0.61857	0.61979
22	0.82835	0.82830
23	0.94753	0.95330
24	0.98619	0.99470
25	0.99258	0.99991
26	0.99258	1.00000

tableau 5.1 Comparaison entre l'approximation $\lambda(x)$ et les valeurs obtenues avec l'approche de Martin pour $F(x)$.

Les résultats sont remarquables. Cependant, ils laissent planer un certain doute sur l'exactitude des résultats de Martin. En effet, la fonction de distribution calculée par Martin n'atteint jamais l'unité et la borne inférieure $\lambda(x)$ est plus élevée que la valeur correspondante de Martin pour $x = 21, 23, 24, 25, 26$. Ces anomalies sont certainement causées par la grande sensibilité des formules de Martin: nous avons noté qu'une petite déviation, même pour la dixième décimale, d'un des coefficients dans les formules de Martin donne des résultats très différents.

L'approximation $\lambda(x)$ pour $F(x)$ en $x = 15, 16, \ldots, 27$ a été obtenue en utilisant l'unité centrale (CPU) durant moins de 0.2 seconde avec Δ égal à 0.1 et K égal à 50 . Nous obtenons la même fonction de distribution approximative pour K aussi grand que 1000 et pour différentes valeurs de Δ autour de 0.1 .

Les bornes pour la moyenne de F ainsi que pour le deuxième moment de F sont:

$$\underline{E[T]} = 20.500 \leq E[T] = 20.524 \leq 20.562 = \overline{E[T]}$$

et

$$\underline{E[T^2]} = 422.250 \leq E[T^2] = 423.139 \leq 424.759 = \overline{E[T^2]}$$

où les valeurs $E[T]$ et $E[T^2]$ sont données par Martin.

Le second exemple auquel nous allons nous attarder fut présenté par Pritsker et Kiviat (1969) pour étudier l'approche de Moder et Phillips lorsqu'on utilise des techniques de simulation pour améliorer la borne du CPM. La figure 5.2 représente le réseau PERT correspondant.

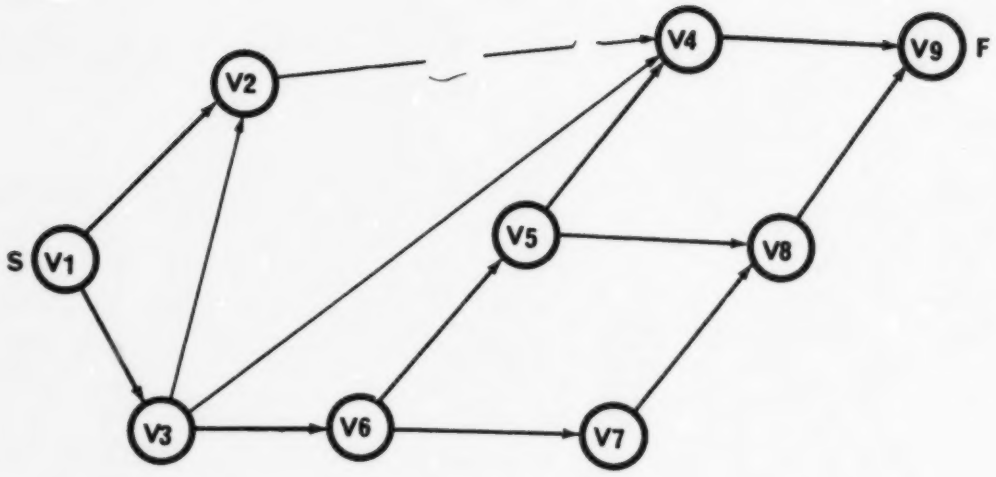


figure 5.2 Exemple d'un réseau PERT proposé par Pritsker et Kiviat.

Les fonctions de distribution des durées des activités sont présumées normales; leurs paramètres sont énumérés au tableau 5.2 .

		durées des activités	
sommet initial	sommet terminal	moyenne	écart type
v1	v2	13.0	3.00
v1	v3	5.5	1.18
v3	v2	7.0	1.00
v3	v4	16.5	2.50
v3	v6	5.2	0.84
v6	v5	6.0	1.00
v6	v7	10.3	1.67
v4	v9	3.2	0.50
v5	v4	20.0	3.32
v5	v8	4.0	0.71
v7	v8	3.2	0.84
v8	v9	16 5	1.18
v2	v4	14.7	1.34

tableau 5.2 Liste des paramètres associés aux fonctions de distribution normale des durées des activités.

Les résultats statistiques obtenus à partir de 400 simulations sont comparés avec les différentes bornes que nous proposons. L'approximation $\lambda(x)$ pour $F(x)$ est présentée au tableau 5.3, parallèlement aux valeurs obtenues avec l'approche de Moder et Philipps améliorée.

x \ F(x)	valeurs de Pritsker et Kiviat	borne inférieure $\lambda(x)$
34	0.0000	0.0000
35	0.0050	0.0000
36	0.0100	0.0000
37	0.0375	0.0000
38	0.0725	0.0000
39	0.1475	0.0000
40	0.2400	0.0000
41	0.3550	0.1136
42	0.4950	0.3825
43	0.6550	0.5963
44	0.7525	0.7532
45	0.8300	0.8586
46	0.9050	0.9237
47	0.9575	0.9609
48	0.9675	0.9808
49	0.9850	0.9910
50	0.9950	0.9959
51	0.9975	0.9983
52	1.0000	0.9993
53	1.0000	0.9997
54	1.0000	1.0000

tableau 5.3 Comparaison entre l'approximation $\lambda(x)$ et les valeurs obtenues par Pritsker et Kiviat.

Les bornes pour la moyenne et le deuxième sommet de F sont:

$$\underline{E[T]} = 40.700 \leq E[T]^{+} = 42.1146 \leq 42.957 = \overline{E[T]}$$

et

$$E[T^2] = 1665.3 \leq E[T^2]^{+} = 1782.3 \leq 1843.9 = \overline{E[T^2]}$$

où les valeurs $E[T]^{+}$ et $E[T^2]^{+}$ sont calculées à partir des résultats de la simulation.

Dans les cas où les fonctions de densité associées aux durées des activités sont discrètes, les bornes proposées par Kleindorfer (1971) peuvent être utilisées. Des études subséquentes devraient indiquer dans quels cas la méthode que nous proposons est plus efficace que celle de Kleindorfer.

Les résultats théoriques et numériques indiquent que cette nouvelle approche est facile d'accès et consomme peu de temps machine. En outre, les résultats sont stables et satisfaisants.

Ces remarques complètent la première partie de ce chapitre. Dans la seconde, nous allons généraliser les résultats obtenus aux réseaux PERT "décisionnels".

5.6 Réseaux PERT "décisionnels"

Nous proposons dans cette seconde partie du chapitre V de généraliser l'utilisation de l'approche probabiliste développée dans les sections précédentes dans les cas où le programme est tel qu'il est impossible de définir exactement le réseau PERT correspondant. Dans plusieurs programmes de recherche ou de développement, certains événements ont une signification très particulière en ce sens que de leur résultat dépend le déroulement ultérieur du programme. Autrement dit, on ne peut établir de programme puisque du résultat de certaines opérations ou de certains événements dépend la conduite du projet. De tels événements sont appelés événements "décisionnels". Un réseau PERT peut admettre des événements "décisionnels" qui sont illustrés sur le graphe par des losanges; un réseau PERT possédant de tels événements est appelé réseau PERT "décisionnel" ou RPD. Un exemple schématique d'un RPD est donné à la figure 5.3. Pospelov et Barishpolets (1966) ont présenté dans un contexte général le concept des réseaux PERT "décisionnels".

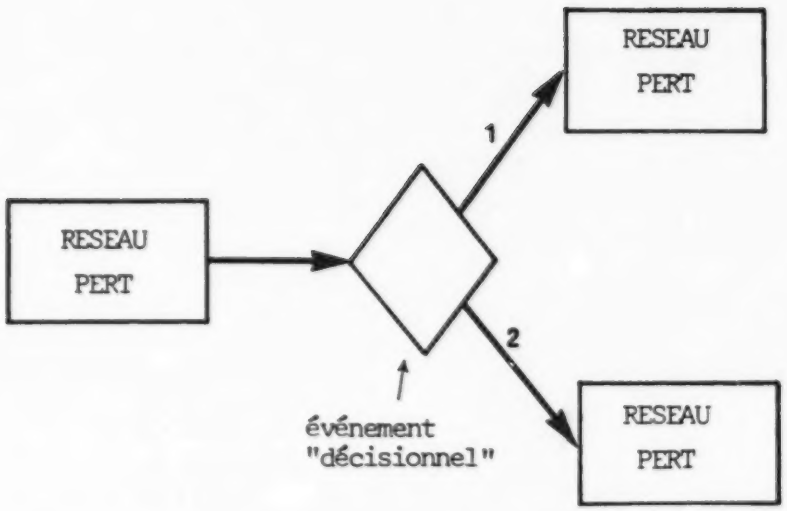


figure 5.3 Exemple schématique d'un RPD (réseau PERT "décisionnel").

La figure 5.4 montre un RPD qui nous servira à illustrer les différentes notions que nous allons introduire d'ici à la fin du chapitre.

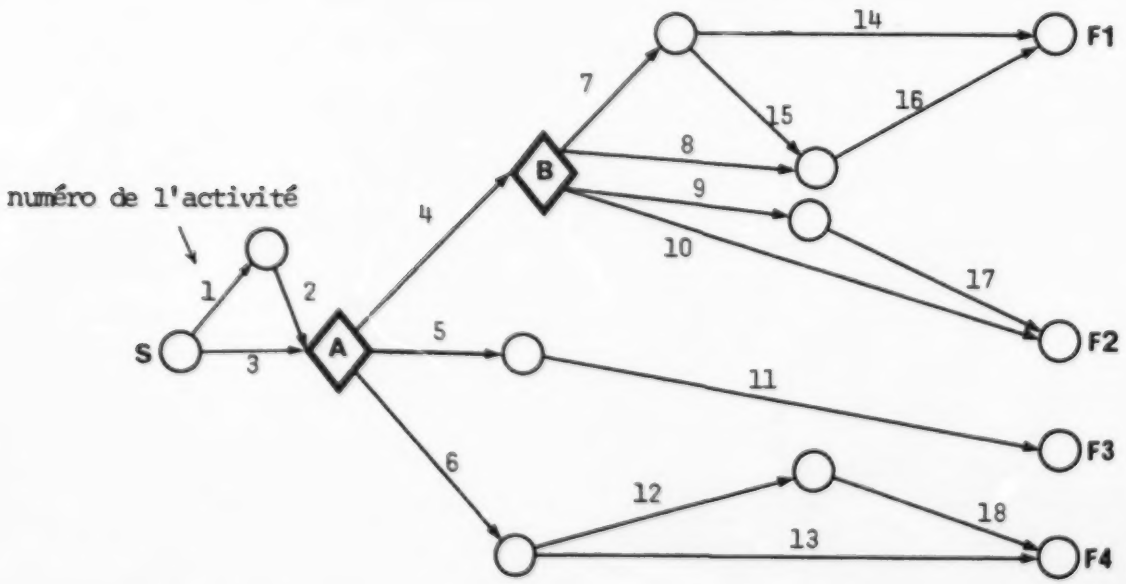


figure 5.4 RPD servant d'exemple.

Le programme correspondant au RPD de la figure 5.4 permet quatre objectifs finals. Nous dénotons par s le nombre de ces objectifs et par $F_i (i = 1,\ldots, s)$ le $i^{ème}$ objectif. Les événements A et B sont des événements "décisionnels".

Un événement "décisionnel" peut donner lieu à plusieurs éventualités. De fait, l'événement "décisionnel" A du RPD exemplaire de la figure 5.4 peut produire trois résultats différents. Il est toujours possible, en ajoutant des arcs et des sommets fictifs, de transformer un RPD de telle sorte que tous les événements "décisionnels" engendrent seulement deux éventualités numérotées 1 et 2. L'éventualité 1 est toujours plus élevée sur le graphe que l'éventualité 2. La figure 5.5 expose le résultat d'une telle transformation pour le RPD de la figure 5.4. Dorénavant, nous ne considérerons que les RPD ayant subi un tel changement.

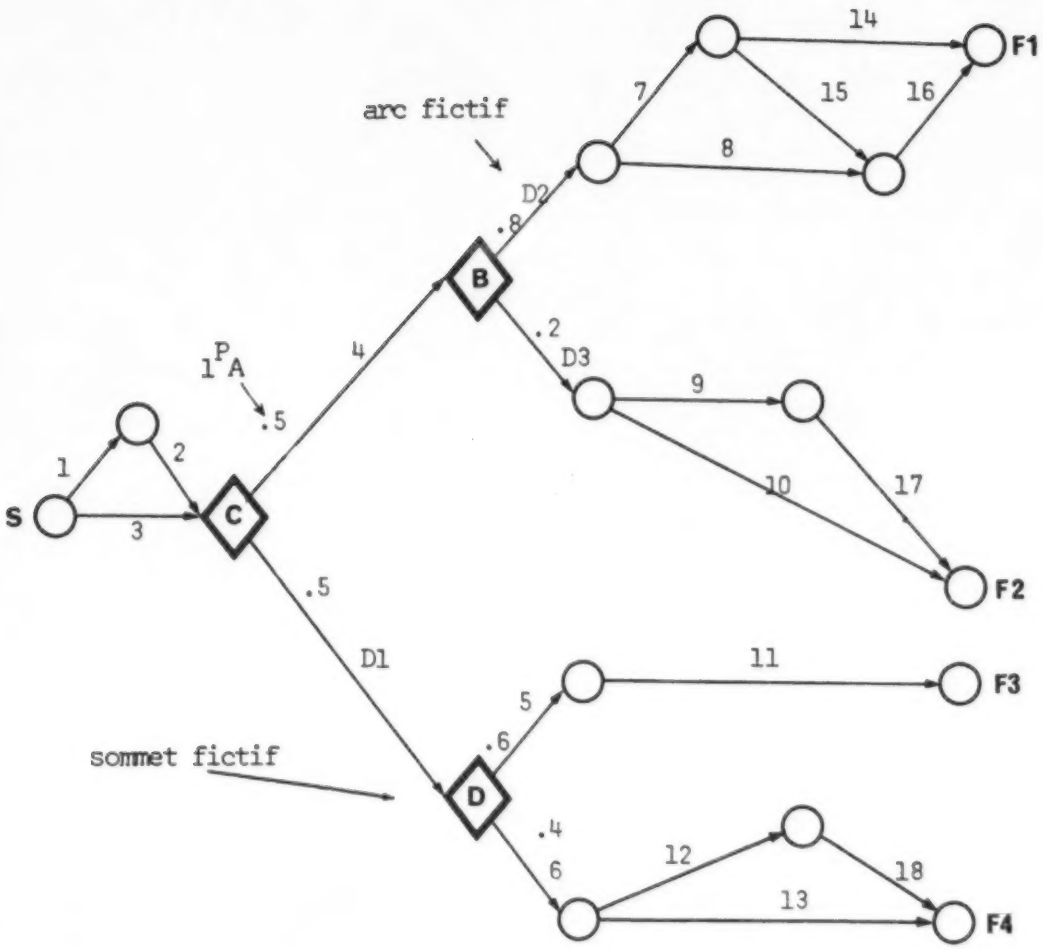


figure 5.5 RPD associé au RPD de la figure 5.4 .

Pour chaque événement "décisionnel" A dans V , P_{iA} (i = 1, 2) dénote la probabilité que l'éventualité i de l'événement "décisionnel" A se produise, étant donné que le sommet A est atteint. Nous avons évidemment la restriction suivante:

$$P_{1A} + P_{2A} = 1 .$$

A la figure 5.5, ces probabilités apparaissent le long des arcs dont le sommet initial est un événement "décisionnel". Ainsi, l'éventualité 1 de l'événement "décisionnel" C se produit avec une probabilité égale à 0.5, si l'on atteint le sommet C .

On convient que les activités possèdent toutes les caractéristiques de celles définies au cours de la première partie de ce chapitre; en particulier, les variables représentant les durées des activités sont des variables aléatoires.

Etant donné un RPD, nous sommes intéressés par les trois questions suivantes:

Quelle est la probabilité que le programme se termine au sommet F_i ?

Quelle est la fonction de distribution approximative de la durée du programme si nous ignorons le résultat final?

Quelle est la fonction de distribution approximative de la durée du programme si l'objectif atteint est F_i ?

Nous allons présenter une approche unifiée pour répondre à ces trois questions lorsque le RPD réduit (RPDR) forme un arbre. Soit G'

le RPDR associé au RPD G . Les sommets de G' sont le sommet-source S , les sommets-objectifs $F_i (i = 1,...,s)$ ainsi que tous les événements "décisionnels". Soit V' l'ensemble des sommets de G' et E' l'ensemble des arcs. Connaissant V' , E' est défini en appliquant la procédure de réduction d'un graphe bien connue dans la théorie des graphes. Ainsi, le graphe G' est complètement défini. Le RPDR du RPD défini à la figure 5.5 est représenté à la figure 5.6. Ce graphe est un arbre.

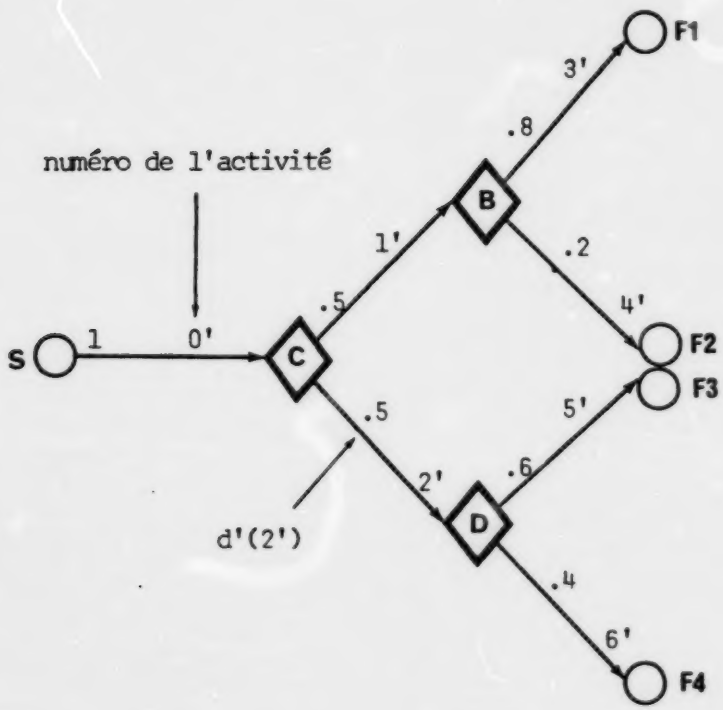


figure 5.6 RPDR associé au RPD de la figure 5.5 .

5.7 Probabilité qu'un sommet soit atteint

Dans G' , on appelle $p_{v'}$ un chemin entre la source S et le sommet v' dans V' . La probabilité d'atteindre le sommet v' dans V' est dénotée par $Pr[v']$.

Nous définissons la fonction $d'(e')$ qui fait correspondre à chaque arc e' dans E' une valeur égale à la probabilité de l'éventualité qui est résumée en e' . Pour l'arc $0'$ dont le sommet initial est S , nous définissons $d'(0')$ égale à 1 . Les différentes valeurs associées aux arcs par la fonction d' apparaissent le long des arcs à la figure 5.6. Dans cet exemple, $d'(1')$ égale 0.5 .

$Pr[v']$ est égale au produit des $d'(e')$ pour lesquelles e' appartient au chemin $p_{v'}$:

$$Pr[v'] = \prod_{E'} d'(e')^{\delta_{e'p_{v'}}}$$

Cette expression est analogue à celle de $\Phi(Z)$ rencontrée au cours de la première partie de ce chapitre où $d'(\cdot)$ remplace $\psi_e(\cdot)$ et où la sommation s'effectue sur un ensemble contenant un seul élément. Utilisant la même procédure de balayage d'un réseau PERT introduite pour évaluer $\Phi(7)$, les $Pr[v']$, pour tout v' dans V' , peuvent être calculées efficacement.

La définition d'un RPDR implique que chaque arc e' dans E' tient la place d'un réseau PERT contenu dans le RPD initial. Ainsi, tous les sommets v dans V fusionnés en e' ont la même probabilité

d'être atteints et cette probabilité égale $Pr[n_2(e')]$. De ce fait, il est possible d'apporter quelques modifications simples et triviales à la procédure de balayage pour évaluer directement sur le RPD initial les $Pr[v]$, v dans V .

5.8 Borne inférieure pour la fonction de distribution de la durée d'un programme représenté par un RPD

Le but de cette section est de répondre aux deuxième et troisième questions.

Nous ajoutons au RPD un sommet fictif F' ainsi que s arcs fictifs reliant $F_i (i = 1, \ldots, s)$ à F' . Nous définissons la fonction $d(e)$ pour tout e dans E de la façon suivante:

$$d(e) = \begin{cases} {}_iP_A & \text{si } n_1(e) \text{ est l'événement "décisionnel" } A \text{ et si } e \text{ est associé à l'éventualité } i \text{ , pour tout } A \text{ appartenant à l'ensemble des événements "décisionnels" et pour } i = 1, 2 \\ 1 & \text{autrement .} \end{cases}$$

Reprenant l'idée émise à la section précédente, nous avons:

$$Pr[F_i] = \prod_E d(e)^{\delta_{ep}} \quad \text{pour } p \text{ appartenant à } P_{F_i}$$

et

$$F(x) = Pr[T \leq x] = \sum_{i=1}^{s} Pr[T \leq x \mid F_i] \, Pr[F_i]$$

où $Pr[T \leq x \mid F_i]$ est la fonction de distribution de la durée du programme sachant qu'il se termine en F_i .

On doit remarquer que la quantité $Pr[T \leq x \mid F_i]$ peut être évaluée directement en utilisant l'approche proposée dans les premières sections de ce chapitre et en ne considérant que le réseau PERT défini par P_{F_i} . Du même coup, nous répondons à la troisième question. Les quantités $Pr[F_i]$ $(i = 1,\ldots s)$ sont évaluées suivant l'approche de la section précédente. D'un point de vue théorique et connaissant $Pr[T \leq x \mid F_i]$ et $Pr[F_i]$ pour $i = 1,\ldots, s$, l'évaluation de $Pr[T \leq x]$ ne pose aucun problème. Nous allons tout de même poursuivre afin de développer une méthode permettant d'obtenir $Pr[T \leq x]$ en balayant une seule fois le RPD.

La fonction de distribution F est telle que:

$$F(x) = \sum_{i=1}^{s} Pr[t_p \leq x \quad \text{pour tout } p \text{ dans } P_{F_i}]\, Pr[F_i] \quad .$$

Une argumentation simple nous permet d'écrire:

$$\max\left\{0,\ 1 - \sum_{i=1}^{s} \sum_{P_{F_i}} Pr[t_p > x]\, Pr[F_i]\right\} \leq \sum_{i=1}^{s} \max\left\{0,\ 1 - \sum_{P_{F_i}} Pr[t_p > x]\right\}$$

$$Pr[F_i] \leq F(x) \ .$$

Nous avons:

$$\sum_{i=1}^{s} \sum_{P_{F_i}} Pr[t_p > x]\, Pr[F_i] = C - \sum_{i=1}^{s} \sum_{P_{F_i}} Pr[F_i]\, F_p(x)$$

où $C = \sum_{i=1}^{s} \Pr[F_i]\, r_{F_i}$.

Empruntant l'argumentation de la section 5.4, nous pouvons écrire:

$$\sum_{i=1}^{s} \sum_{P_{F_i}} \Pr[F_i]\, F_p(x) = C \int_x^{-1} \left[\frac{\sum_{i=1}^{s} \sum_{P_{F_i}} \Pr[F_i]\, \psi_p(Z)}{C} \right]$$

et

$$\sum_{i=1}^{s} \sum_{P_{F_i}} \Pr[F_i]\, \psi_p(Z) = \sum_{P_F} \prod_{E} [d(e)\, \psi_e(Z)]^{\delta_{ep}} = \phi^{+}(Z) \quad .$$

L'approximation $\lambda^{+}(x)$ pour $F(x)$ est donnée par:

$$\lambda^{+}(x) = \max \left\{ 0,\ 1 - C + C \int_x^{-1} \left[\frac{\phi^{+}(Z)}{C} \right] \right\} \leq F(x) \quad .$$

Cette borne est comparable à celle obtenue dans le cas des réseaux PERT simples. Utilisant la formule de Davies, nous pouvons écrire:

$$\lambda^{+}(x) = \max \left\{ 0,\ 1 - \frac{\sum_{i=1}^{s} \Pr[F_i]\, r_{F_i}}{2} - \sum_{k=0}^{K} \frac{\mathrm{IM}\left[\phi^{+}((k + 0.5)\Delta)\, e^{-i(k + 0.5)\Delta x} \right]}{\pi\,(k + 0.5)} \right\}$$

$$\leq F(x) \quad .$$

En prenant la borne $\lambda^{+}(x)$, il est possible d'obtenir des bornes pour les différents moments de la fonction de distribution $F(x)$ en accord

avec la théorie exposée à la section 5.4 .

Nous avons ainsi développé une méthode offrant l'opportunité de répondre à trois questions fondamentales posées par les organisateurs de programmes et ce, en balayant une seule fois les RPD associés aux programmes.

CHAPITRE VI

DES BORNES INFERIEURES POUR LA DUREE MOYENNE D'UN PROGRAMME

6.1 Notation et définitions

Soit un réseau PERT G composé de n sommets et de m arcs. Les sommets sont dénotés par l'indice i(i= 1,..., n) de sorte que 1 est le sommet-source S et n le sommet-objectif F ; s'il existe un arc allant du sommet j au sommet k , j est alors inférieur à k . Les arcs sont identifiés par la lettre e (e= 1,..., m) . θ_i réunit les arcs e pour lesquels $n_2(e) \leq i$:

$$\theta_i = \{e \text{ dans } E \mid n_2(e) \leq i\} .$$

β_i est un sous-ensemble de θ_i défini de la façon suivante:

$$\beta_i = \{e \text{ dans } \theta_i \mid n_2(e) = i\} .$$

De plus, $\beta_i \cup \overline{\beta}_i = \theta_i$. E_i et $\overline{E}_i$ sont deux sous-ensembles disjoints de θ_i satisfaisant la contrainte $E_i \cup \overline{E}_i = \theta_i$.

Nous supposons que l'ensemble des temps requis pour accomplir les activités associées aux arcs entrant dans un sommet est indépendant de l'ensemble des temps associés aux arcs entrant dans un autre sommet. En outre, nous supposons que les temps requis pour accomplir les activités sont des variables aléatoires appelées t_e , discrètes ou continues et définies sur un ensemble positif borné.

Si θ est un sous-ensemble quelconque de E, il advient que T_θ réunit l'ensemble des variables aléatoires t_e pour lesquelles e appartient à θ :

$$T_\theta = \{t_e \mid e \text{ dans } \theta \subseteq E\} .$$

La durée d'un programme jusqu'au sommet i est dénotée par ℓ_i. Par conséquent, ℓ_1 égale 0 et ℓ_n est le temps nécessaire pour exécuter tout le programme.

Nous allons maintenant énoncer une proposition fondée sur l'inégalité de Jensen reprise par Rao (1971) qui nous sera utile par la suite.

Proposition A

Si $X_1 \ldots X_m$ sont des variables aléatoires,

$$E_{X_1 \ldots X_m}\left[\max_{j \text{ dans } \{1,\ldots\}}\left\{f_j(X_1,\ldots,X_m)\right\}\right] \leq \max_{j \text{ dans } \{1,\ldots\}}\left\{E_{X_1 \ldots X_m}\left[f_j(X_1,\ldots,X_m)\right]\right\}$$

où $E_{X_1 \ldots X_m}$ est l'espérance mathématique impliquant les variables aléatoires $X_1 \ldots X_m$ et où f_j est une fonction définie sur ces variables aléatoires.

6.2 Énumération de certaines bornes inférieures pour la durée moyenne d'un programme

Nous nous intéressons au temps moyen requis pour exécuter toutes les activités d'un programme; en particulier, nous présentons une méthode en vue d'obtenir des bornes inférieures pour ce temps moyen. La

définition même de ℓ_i nous permet d'écrire pour $1 < i \leq n$ la relation suivante:

$$\ell_i = \max_{\beta_i} \left\{ \ell_{n_1(e)} + t_e \right\} .$$

En prenant l'espérance mathématique du terme de gauche et du terme de droite, nous obtenons:

$$E[\ell_i] = \underset{T_{\theta_i}}{E} [\ell_i] = \underset{T_{\theta_i}}{E} \left[\max_{\beta_i} \left\{ \ell_{n_1(e)} + t_e \right\} \right]$$

où $E[\ell_i]$ représente le temps moyen requis pour atteindre l'événement i du programme.

Si nous divisons l'ensemble T_{θ_i} en deux parties, nous avons:

$$E[\ell_i] = \underset{T_{E_i}}{E} \; \underset{T_{\overline{E}_i}}{E} \left[\max_{\beta_i} \left\{ \ell_{n_1(e)} + t_e \right\} \right].$$

La Proposition A permet d'écrire l'inégalité suivante:

$$E[\ell_i] \geq \underset{T_{E_i}}{E} \left[\max_{\beta_i} \left\{ \underset{T_{\overline{E}_i}}{E} \left[\ell_{n_1(e)} + t_e \right] \right\} \right].$$

Cette inégalité donne une borne inférieure générale pour $E[\ell_i]$. Nous allons considérer quelques cas particuliers de cette borne à partir de différentes définitions de l'ensemble E_i .

a) $E_i = \phi$ l'ensemble vide

Si E_i égale l'ensemble vide, la borne générale se ramène à

l'inégalité:

$$E[\ell_i] \geq \max_{\beta_i} \left\{ E_{T_{\theta_i}} [\ell_{n_1(e)}] + E_{T_{\theta_i}} [t_e] \right\} \quad \text{pour} \quad 1 < i \leq n \ .$$

Si nous définissons récursivement

$$g_1 = 0$$

et

$$g_i = \max_{\beta_i} \left\{ g_{n_1(e)} + E_{t_e} [t_e] \right\} \quad \text{pour} \quad 1 < i \leq n \ ,$$

$g_i (i = 1, \ldots, n)$ représente alors la borne familière obtenue par la méthode CPM lorsque les t_e sont remplacées par les $E[t_e]$. Il est facile de démontrer que g_i est une borne inférieure pour $E[\ell_i]$:

$$g_i \leq E[\ell_i] \quad \text{pour} \quad 1 \leq i \leq n \ .$$

En effet, cette relation est vérifiée pour i égal à 1 et nous n'avons qu'à poursuivre par induction pour finir la preuve.

b) $E_i = \beta_i$

Pour E_i défini comme l'ensemble des arcs entrant dans le sommet i , l'hypothèse d'indépendance nous permet d'obtenir la borne inférieure:

$$E[\ell_i] \geq E_{T_{\beta_i}} \left[\max_{\beta_i} \left\{ E_{T_{\bar{\beta}_i}} [\ell_{n_1(e)}] + t_e \right\} \right] \quad \text{pour} \quad 1 < i \leq n$$

où $\bar{\beta}_i = \{e \text{ dans } E \mid n_2(e) < i\}$.

Si nous définissons récursivement

$$f_1 = 0$$

et

$$f_i = E_{T_{\beta_i}}\left[\max_{\beta_i} \left\{ f_{n_1(e)} + t_e \right\} \right] \quad \text{pour} \quad 1 < i \leq n \ ,$$

$f_i (i = 1, \ldots, n)$ représente alors la borne de Fulkerson pour les cas discrets ou continus. La borne proposée par Elmaghraby qui améliore celle de Fulkerson s'obtient presque directement en inversant le sens des arcs appartenant à θ_i . Encore une fois, nous pouvons démontrer facilement par induction que:

$$f_i \leq E[\ell_i] \quad \text{pour} \quad 1 \leq i \leq n \ .$$

En appliquant la Proposition A, nous avons:

$$f_i \geq \max_{\beta_i} \left\{ f_{n_1(e)} + E[t_e] \right\}$$

et il s'ensuit que $g_i \leq f_i$ pour $1 \leq i \leq n$, comme prévu.

c) $E_i = \theta_i$

Si E_i égale θ_i , la borne inférieure atteint la durée moyenne du programme, c'est-à-dire que la borne inférieure est parfaite:

$$E[\ell_i] \geq E_{T_{E_i}}\left[\max_{\beta_i} \left\{ \ell_{n_1(e)} + t_e \right\} \right] = E_{T_{\theta_i}} [\ell_i] \ .$$

Comme nous l'avons mentionné, l'évaluation de $E[\ell_i]$ demande malheureusement une énorme quantité de travail.

Il s'agit de trouver l'équilibre entre l'effort que l'on est prêt à fournir et la qualité de la borne que l'on désire.

Nous pouvons réunir en E_i tous les arcs de θ_i jugés pertinents, en autant que la définition de E_i qui en découle permet d'évaluer de façon efficace une borne satisfaisante pour $E[\ell_i]$. Dans la section suivante, une nouvelle borne est introduite suivant ce principe.

6.3 Elaboration de nouvelles bornes

Nous avons l'intention de généraliser la borne proposée par Fulkerson et à cette fin, nous introduisons quelques définitions.

Soit p un chemin dans G . Nous étendons la définition de la fonction n à tout p dans G :

$n(p) = (j,k) = [n_1(p) , n_2(p)]$ pour tout p dans G où j est le sommet initial du chemin p et k , son sommet terminal. A chaque p dans G , nous associons une variable aléatoire t_p qui indique le temps nécessaire pour accomplir séquentiellement les activités constituant le chemin p . Nous dénotons par $d^+(j)$ le nombre d'arcs entrant dans le sommet j et par $d^-(j)$ le nombre d'arcs en sortant.

Nous allons nous concentrer sur le sommet i, $1 < i \leq n$. Soit $P.i$ un ensemble regroupant un certain nombre de chemins dans G se terminant au sommet i pour lequel ensemble nous pouvons écrire:

$$\ell_i = \max_{P.i} \left\{ \ell_{n_1(p)} + t_p \right\} .$$

Entre autres, P_i et β_i répondent à la définition de P.i . Aussi, en ce qui concerne le sommet i , au lieu d'exprimer ℓ_i à partir des sommets $n_1(e)$ pour e dans β_i , c'est-à-dire les sommets qui précèdent immédiatement le sommet i , nous remontons jusqu'au sommet $n_1(p)$ pour p dans P.i . Il faut noter que β_i est un sous-ensemble de P.i . Cette nouvelle relation amène l'inégalité qui suit; cette dernière tient lieu de borne inférieure généralisée pour $E[\ell_i]$:

$$E[\ell_i] \geq \underset{T_{E_i}}{E}\left[\max_{P.i}\left\{\underset{T_{E_i}}{E}\left[\ell_{n_1(p)} + t_p\right]\right\}\right].$$

Pour obtenir une borne particulière, nous devons spécifier les ensembles P.i et E_i . p défini par les sommets $j = j_1, \ldots, j_\ell = i$ appartient à l'ensemble P.i si

$$d^+(j_k) = d^-(j_k) = 1 \quad \text{pour} \quad k = 2, \ldots, \ell - 1 \ ,$$

si

$$d^+(i) = 1$$

et si

$$d^-(j) \neq 1 \ .$$

Tous les sommets appartenant à un chemin p dans P.i ont seulement un arc entrant et un arc sortant, sauf le sommet $n_1(p)$ qui n'a pas d'arc entrant ou qui en a plus qu'un et le sommet $n_2(p)$ dont le nombre d'arcs sortant n'est soumis à aucune contrainte. E_i est défini comme suit:

$$E_i = \{e \text{ dans } \theta_i \mid e \text{ appartient à } p \text{ dans } P.i\} \ .$$

Ce sont deux définitions possibles pour les ensembles P.i et E_i . Il demeure toutefois possible d'étudier d'autres configurations.

Si nous définissons récursivement

$$m_1 = 0$$

et

$$m_i = \underset{T_{E_i}}{E}\left[\max_{P.i} \left\{ m_{n_1(p)} + t_p \right\} \right] \text{pour } 1 < i \leq n ,$$

nous faisons face à une généralisation de la borne de Fulkerson. Pour un m_i particulier, nous devons évaluer la durée moyenne d'un sous-programme représenté par un réseau PERT réduit et doté d'une structure singulière: en effet, tous les chemins du sommet-source au sommet-objectif sont parallèles. La figure 6.1 illustre un tel réseau PERT où les variables aléatoires, associées aux chemins, apparaissent le long du dernier arc de chaque chemin. q indique le nombre de chemins dans P.i . Les sommets $D_0, D_1, \ldots, D_q$ sont des sommets fictifs et les arcs (D_0, D_j) , $1 \leq j \leq q$, sont aussi des arcs fictifs dont la durée est une constante égale à $m_{n_1(p_j)}$.

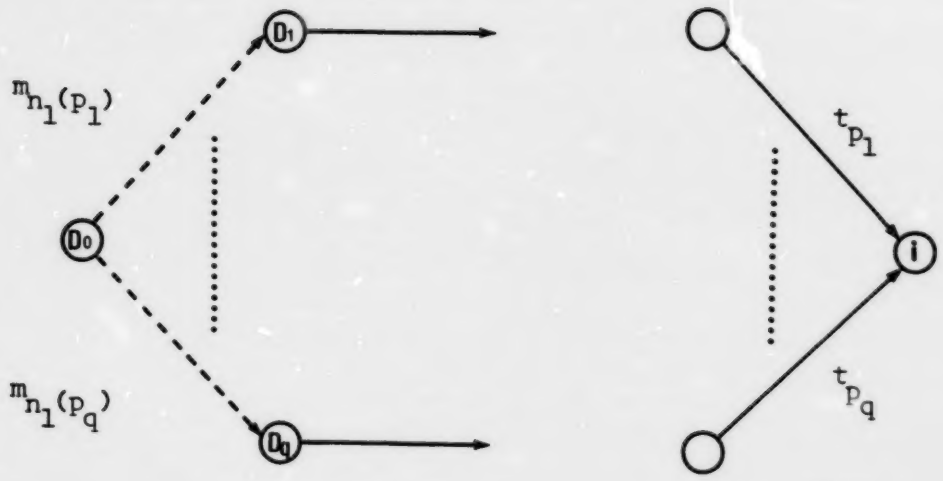


figure 6.1 Exemple d'un réseau PERT parallèle.

Soit T' le temps nécessaire pour atteindre le sommet-objectif i dans un réseau PERT parallèle:

$$T' = \max_{P.i} \left\{ m_{n_1(p)} + t_p \right\} .$$

Si la fonction de densité de T' est facilement accessible, nous pouvons obtenir directement m_i . Toutefois, cette situation est pour le moins inhabituelle. Cependant, la méthode de Martin (1965) qui évalue la fonction de densité de la durée d'un programme lorsque les fonctions de densité associées aux arcs du réseau PERT s'expriment sous forme polynômiale, peut être très utile en regard de la structure particulière du réseau PERT sous-jacent au problème qui nous concerne. En effet, selon la notation

de Martin, nous éliminons la première étape qui consiste à transformer un réseau PERT en un réseau PERT série-parallèle. De plus, la réduction du réseau PERT série-parallèle revient à transformer un réseau PERT parallèle contenant q chemins en un arc équivalent dont la durée associée est égale à T'. Aussi, quand les fonctions de densité associées aux arcs ont une forme polynômiale, l'évaluation de m_i est presque immédiate. La classe des fonctions de densité pouvant être exprimées par des polynômes fournit un choix assez vaste de fonctions: la distribution uniforme et la distribution bêta avec paramètres entiers en sont des exemples. Evidemment, nous avons:

$$f_i \leq m_i \leq E[\ell_i] \quad .$$

Plus généralement, il est difficile d'évaluer m_i parce que nous devons considérer plusieurs t_p à la fois. Cependant, il est possible d'obtenir une borne n_i satisfaisant :

$$f_i \leq n_i \leq E[\ell_i] \qquad 1 \leq i \leq n$$

pour laquelle nous n'avons à considérer que les t_e. Nous terminerons ce chapitre en donnant trois définitions possibles pour n_i.

En premier lieu, nous remplaçons toutes les variables aléatoires t_e, pour e dans E_i, par leur moyenne sauf pour les arcs dont $n_2(e) = i$. Ensuite, nous utilisons la borne de Fulkerson sur le réseau PERT parallèle. Si nous dénotons la valeur résultante par n_i^1, nous avons $f_i = n_i^1$.

La deuxième définition est fondée sur l'observation suivante: dans un réseau PERT parallèle, si nous permutons les arcs le long d'un chemin, la durée moyenne du sous-programme ne change pas. Cependant, si nous calculons n_i^1 pour différentes permutations, n_i^1 peut varier. Pour le réseau PERT parallèle de la figure 6.2 où la variable aléatoire t_e prend les valeurs apparaissant le long de l'arc e avec équiprobabilité, n_i^1 égale 6.5. Si nous permutons les arcs (1,3) et (3,i), n_i^1 égale 6.75. Enfin, si nous permutons les arcs (1,3) et (3,i) ainsi que les arcs (1,2) et (2,i), n_i^1 égale 6.16. Nous choisissons pour n_i^2 la valeur maximale prise par n_i^1 pour toutes les configurations possibles. Nous avons la relation:

$$f_i \leq n_i^2 \leq m_i \leq E[\ell_i] \quad .$$

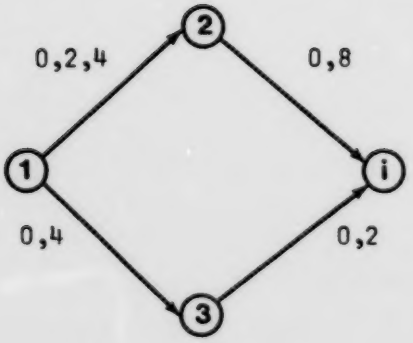


figure 6.2 Exemple d'un réseau PERT parallèle sur lequel on calcule n_i^2 .

La troisième définition, produisant la borne n_i^3 , utilise la borne proposée par Elmaghraby (1967) fondée sur l'observation qu'il est possible

d'inverser le sens des arcs d'un réseau PERT sans changer la durée moyenne du programme; cependant, à chaque étape nous remplaçons f_i par n_i^2. Ainsi:

$$f_i \leq s_i \leq n_i^3 \leq E[\ell_i]$$

où s_i représente la borne initiale d'Elmaghraby. Il est à noter que n_i^3 peut dépasser m_i.

Parmi toutes les bornes inférieures présentées et parmi toutes celles qui émergent de façon naturelle, nous devons choisir celle qui répond le mieux à nos critères de précision et de coût. Les choses en sont presque toujours ainsi: plus les résultats sont précis et plus les coûts sont grands. En ce qui nous concerne, les coûts s'expriment en termes de calculs supplémentaires.

CONCLUSION

Nous avons étudié deux problèmes précis faisant appel aux modèles probabilistes appliqués à la théorie des réseaux. Les résultats obtenus prouvent que nous avons atteint les objectifs que nous nous étions fixés.

L'étendue du champ de recherche offert par les problèmes de transport et d'ordonnancement nous a amené à considérer divers sujets. Afin de permettre une meilleure compréhension, nous avons jugé bon de circonscrire ces sujets à l'intérieur de six chapitres. Ainsi, une discussion sur l'apport de nos résultats de même que des conclusions suivent chacun de ces chapitres.

Somme toute, en ce qui a trait au transport, nous avons facilité l'accès à l'analyse de l'affectation des voyages en améliorant l'algorithme d'affectation probabiliste sans capacité. Nous avons collaboré à mieux comprendre le phénomène du transport et à trouver des solutions adaptées à la réalité en développant un outil qui permet une étude dans le temps du comportement des voyageurs. En outre, nous avons indiqué la façon de substituer à l'enquête origine-destination utilisée lors de la distribution des voyages, des données de trafic pouvant être obtenues simplement. D'un point de vue économique, cette substitution engendre des épargnes appréciables alors que d'un point de vue conceptuel, elle permet l'analyse de la génération, de la distribution et de l'affectation des voyages de façon intégrée.

En ce qui concerne l'ordonnancement, nous avons présenté une façon d'estimer la fonction de distribution de la durée d'un programme ainsi

que de ses différents moments. Les résultats théoriques et numériques témoignent du fait que cette nouvelle approche est facile d'accès et consomme peu de temps machine. Une généralisation immédiate permet pour la première fois une étude efficace de certains réseaux PERT "décisionnels". Enfin, nous avons proposé une nouvelle approche pour obtenir des bornes inférieures à la durée moyenne d'un programme et certaines d'entre elles améliorent les bornes existant déjà.

Au terme de cette étude où ont été exposés différents aspects des problèmes de transport et d'ordonnancement, il serait bon d'envisager les voies de développement qui s'offrent aux solutions que nous avons proposées.

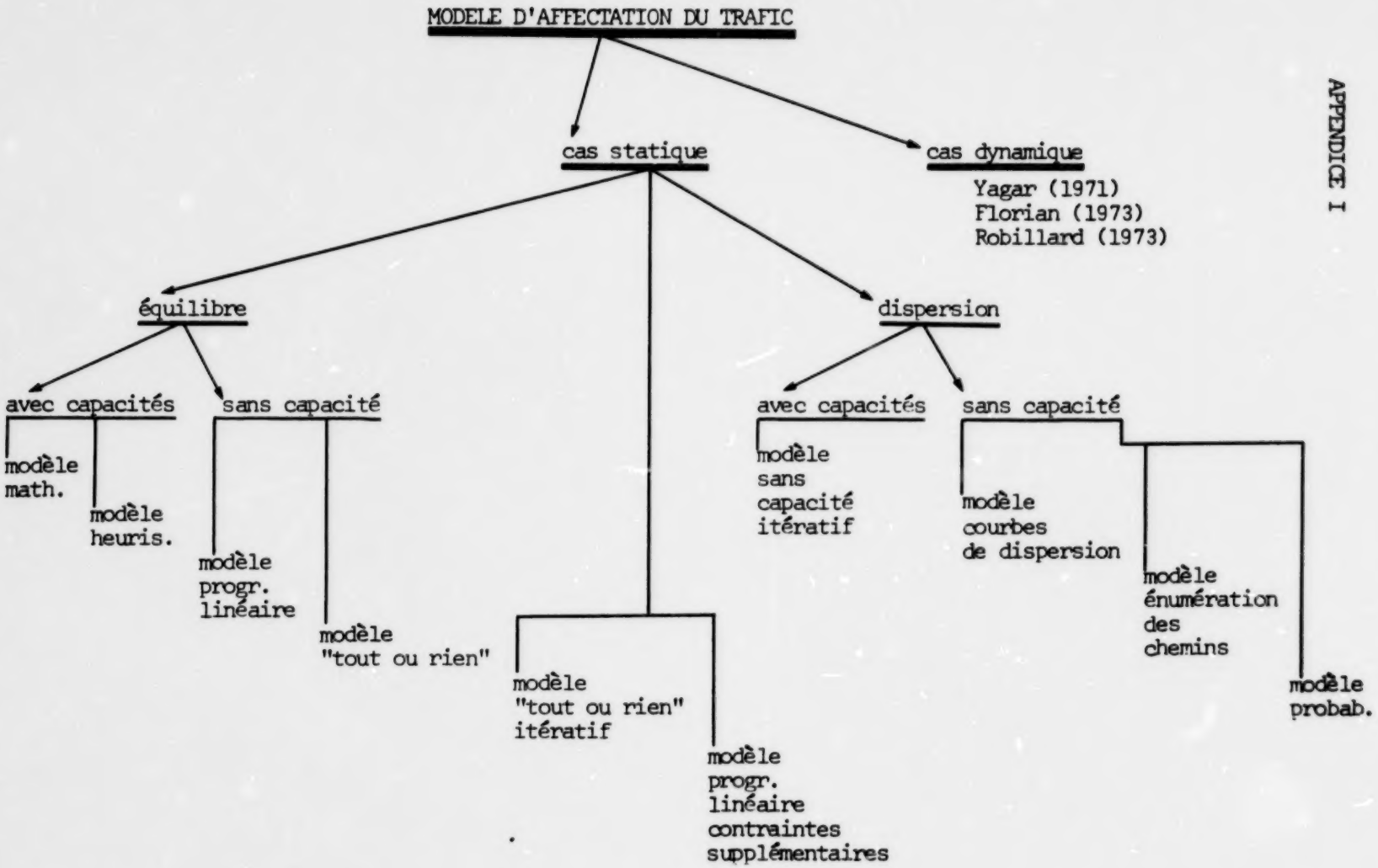
Ayant réalisé le développement théorique de différents modèles de distribution et d'affectation des voyages et ayant effectué quelques essais sur ordinateur du modèle d'affectation probabiliste, il importe de passer à l'expérimentation de ces modèles, de les confronter à la réalité, de poursuivre leur mise en application, de les raffiner et d'étendre leur champ d'action.

Pour ce qui est des problèmes d'ordonnancement, nous avons considéré la notion d'événement "décisionnel" dans les réseaux PERT. Ce travail est une première tentative pour insérer dans l'analyse des programmes un aspect important de la réalité. Nous croyons qu'il reste beaucoup à faire dans ce domaine de la recherche, à commencer par l'introduction des concepts de la programmation dynamique.

De plus, il reste à aborder le problème du choix du réseau PERT lorsque plusieurs configurations sont possibles de même que le problème de l'allocation optimale des ressources budgétaires entre les différentes

activités. En effet, puisque chaque activité consomme des biens, il faudrait à l'avenir envisager le problème de l'affectation optimale en biens entre les différentes activités du programme et cela, en tenant compte des contraintes temporelles et techniques. D'ailleurs, ce sont le plus souvent ces contraintes qui se posent en obstacle dans les problèmes concrets.

Finalement, nous croyons que la procédure de balayage d'un réseau utilisée aux chapitres II, III et V peut servir pour plusieurs autres problèmes dont celui de l'évaluation du flot maximal dans les réseaux probabilistes.


MODELE D'AFFECTATION DU TRAFIC
cas statique
cas dynamique
Yagar (1971)
Florian (1973)
Robillard (1973)
équilibre
dispersion
avec capacités
sans capacité
avec capacités
sans capacité
modèle math.
modèle heuris.
modèle progr. linéaire
modèle "tout ou rien"
modèle sans capacité itératif
modèle courbes de dispersion
modèle énumération des chemins
modèle probab.
modèle "tout ou rien" itératif
modèle progr. linéaire contraintes supplémentaires

APPENDICE II

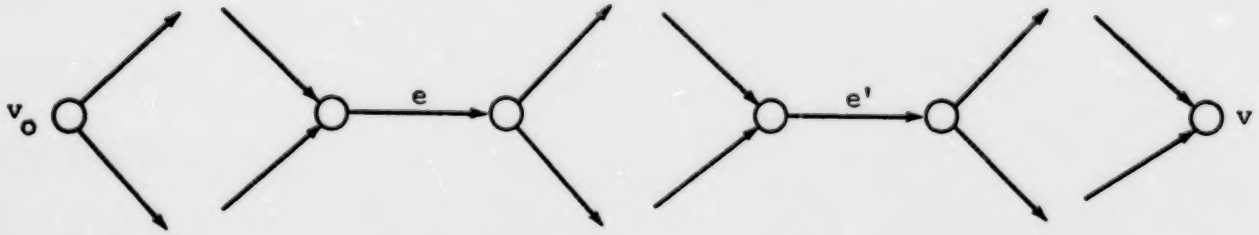
ETUDE DU TERME $\sum_{E} \sum_{e' \neq e \text{ dans } E} \mu_{e|n}\, \mu_{e'|n} \sum_{P_v} \delta_{ep}\, \delta_{e'p}\, Pr(p|j)$

La difficulté réside dans l'évaluation de la sommation:

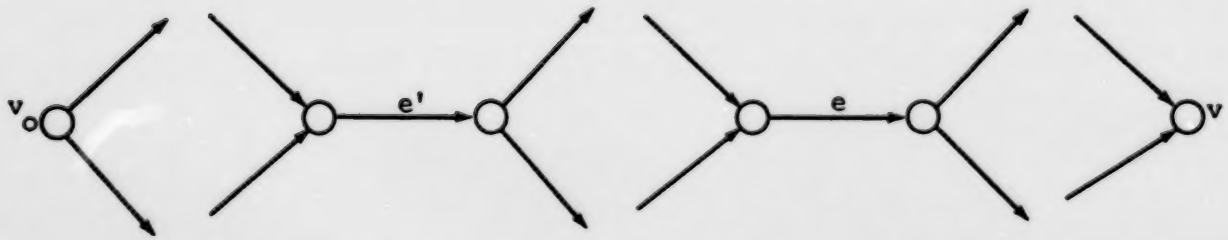
$$\sum_{P_v} \delta_{ep}\, \delta_{e'p}\, Pr(p|j) \quad .$$

Aussi, allons-nous regarder ce terme plus attentivement.

$\sum_{P_v} \delta_{ep}\, \delta_{e'p}\, Pr(p|j)$ représente la probabilité d'emprunter un chemin passant par les arcs e et e' pour quiconque part du sommet v_o au temps n tel que n appartient à $(a_j, a_{j+1}]$. Un tel chemin doit avoir l'allure suivante:



ou



Donc, pour $e = (k,\ell)$ et $e' = (s,y)$,

$$\sum_{P_v} \delta_{ep}\, \delta_{e'p}\, Pr(p|j)$$

$$= \frac{1}{w(v_o,v,j)} \Big[\; w(v_o,k,j)\; d(e,j)\; w(\ell,s,j)\; d(e',j)\; w(y,v,j)$$

$$+ \; w(v_o,s,j)\; d(e',j)\; w(y,k,j)\; d(e,j)\; w(\ell,v,j) \Big]$$

$$= \frac{d(e,j)\; d(e',j)}{w(v_o,v,j)} \Big[\; w(v_o,k,j)\; w(\ell,s,j)\; w(y,v,j) + w(v_o,s,j)\; w(y,k,j)\; w(\ell,v,j) \Big] \; .$$

Le coût d'évaluation de ce terme demeure tout de même élevé.

APPENDICE III

MODE D'UTILISATION DU PROGRAMME REPERT: REseau PERT

III.1 Sommaire

Le but de cet appendice est de fournir divers renseignements préalables à l'utilisation du programme REPERT qui exploite les améliorations techniques développées au chapitre V. REPERT permet l'étude de la fonction de distribution associée au temps nécessaire pour réaliser partiellement ou totalement un programme. Il est entendu que le temps nécessaire à l'accomplissement de chacune des activités associées à ce programme est une variable aléatoire. Ces variables aléatoires sont indépendantes les unes des autres et définies sur un ensemble positif borné. A chaque événement est associée une variable aléatoire représentant le temps nécessaire pour atteindre cet événement.

Bref, le programme REPERT accomplit le travail suivant: dans un premier temps, il lit un bloc de données dont il extrait un réseau PERT. Dans un deuxième temps, REPERT reçoit par l'entremise du fichier d'entrée la liste des événements du programme dont l'usager veut étudier la fonction de distribution. REPERT produit alors une borne inférieure pour chacune des fonctions de distribution et des bornes supérieures pour les deux premiers moments de ces fonctions de distribution. Dans cet appendice, nous trouvons une discussion concernant la façon d'appliquer certains éléments théoriques sous-jacents à REPERT, une description

du programme lui-même, un exemple ainsi que les énoncés en FORTRAN EXTENDED du programme REPERT. Ainsi, par l'entremise du programme REPERT, il sera possible de rendre évidents les avantages et les inconvénients de cette nouvelle approche présentée au chapitre V comparativement à ceux des approches déjà expérimentées.

III.2 Mise en place de REPERT

La mise en place sur ordinateur de la théorie présentée au chapitre V exige la manipulation en mémoire d'un réseau PERT ainsi que le développement d'un algorithme efficace permettant l'évaluation de la fonction Φ en plusieurs points lors d'un balayage du réseau PERT. Cette section porte sur ces deux éléments de REPERT.

a) Représentation d'un réseau PERT sur ordinateur

Plusieurs représentations d'un réseau PERT sur ordinateur sont possibles. Nous allons utiliser celle expérimentée par Lefebvre (1971). Les éléments du réseau PERT sont représentés par des blocs de mots-mémoire. Ces blocs sont reliés de façon qu'on puisse construire en quelque sorte les connexions du réseau PERT. On distingue deux sortes de blocs: les blocs-sommets et les blocs-arcs.

La représentation d'un arc comprend huit éléments dont quatre servent au chaînage:

- l'adresse relative du bloc-sommet associé au point d'origine de l'arc, dénotée par ORIG à la figure III.1;

- l'adresse relative du bloc-sommet associé au point de destination de l'arc, DEST ;
- l'adresse relative d'un autre bloc-arc touchant au point d'origine, P1 ;
- l'adresse relative d'un autre bloc-arc touchant au point de destination, P2 ;
- le type de la fonction de distribution de l'arc, IFCTDIS ;
- le premier paramètre de la fonction de distribution, PARA1 ;
- le second paramètre de la fonction de distribution, PARA2 ;
- enfin, un espace libre pour utilisation ultérieure, LIBRE .

La représentation d'un sommet comprend cinq éléments dont deux servent au chaînage:

- l'adresse relative d'un autre bloc-sommet, PN ;
- l'adresse relative d'un bloc-arc qui touche à ce sommet, PA ;
- un nom par lequel l'usager fait référence au sommet, ID ;
- le nombre d'arcs entrant dans ce sommet, ARCENT ;
- enfin, une zone contenant les $K + 1$ valeurs complexes de la fonction Φ_{ID} , $\Phi_{ID}(Z)$.

Sur l'ordinateur CDC CYBER 74 de l'Université de Montréal où les mots-mémoire comptent 60 bits, on peut représenter les différents blocs comme l'illustre la figure III.1 .

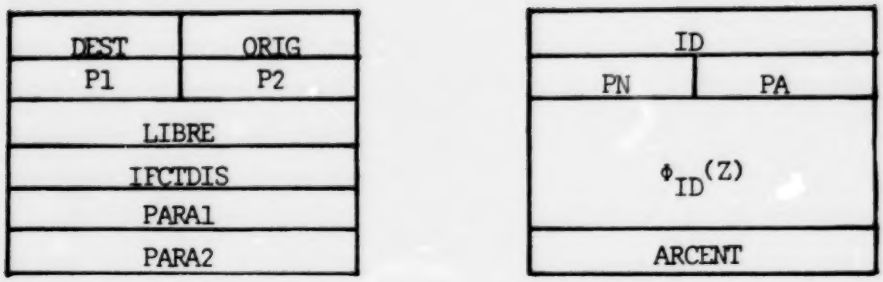


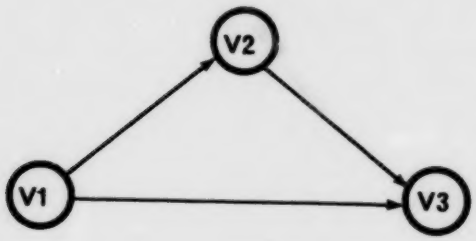
figure III.1 Structure utilisée pour représenter les éléments du réseau PERT.

Ainsi, un bloc-arc occupe 6 mots-mémoire et un bloc-sommet occupe (2K + 5) mots-mémoire; K est défini au chapitre V lors de l'introduction de l'approximation de Davies. On remarque que l'usager peut utiliser 10 caractères pour identifier un sommet et que les demi-mots de 30 bits contiennent des adresses relatives donnant une possibilité de mémoire virtuelle égale à 2^{29} - 1 mots. Toutefois, dans la version présentée ici, REPERT exige que tout le réseau PERT tienne en mémoire centrale, car les blocs sont empilés dans un immense vecteur entier appelé RESO. Ce vecteur est en équivalence avec le vecteur ZESO pour les mots contenant des nombres réels, comme PARA1 et PARA2.

A partir d'un sommet, on peut atteindre tout autre sommet; on peut aussi atteindre tout arc touchant au sommet selon que cet arc se

dirige vers le sommet ou qu'il s'en éloigne. Les avantages de cette représentation sont l'accès facile à un arc ou à un sommet, l'espace-mémoire économisé et variable ainsi que le caractère complet d'un bloc.

La figure III.3 donne le contenu des différents blocs pour le réseau PERT de la figure III.2 .



F_{V1V2} = uniforme (0,1)

F_{V1V3} = normale tronquée (10,5)

F_{V2V3} = bêta (1,1)

figure III.2 Exemple d'un réseau PERT.

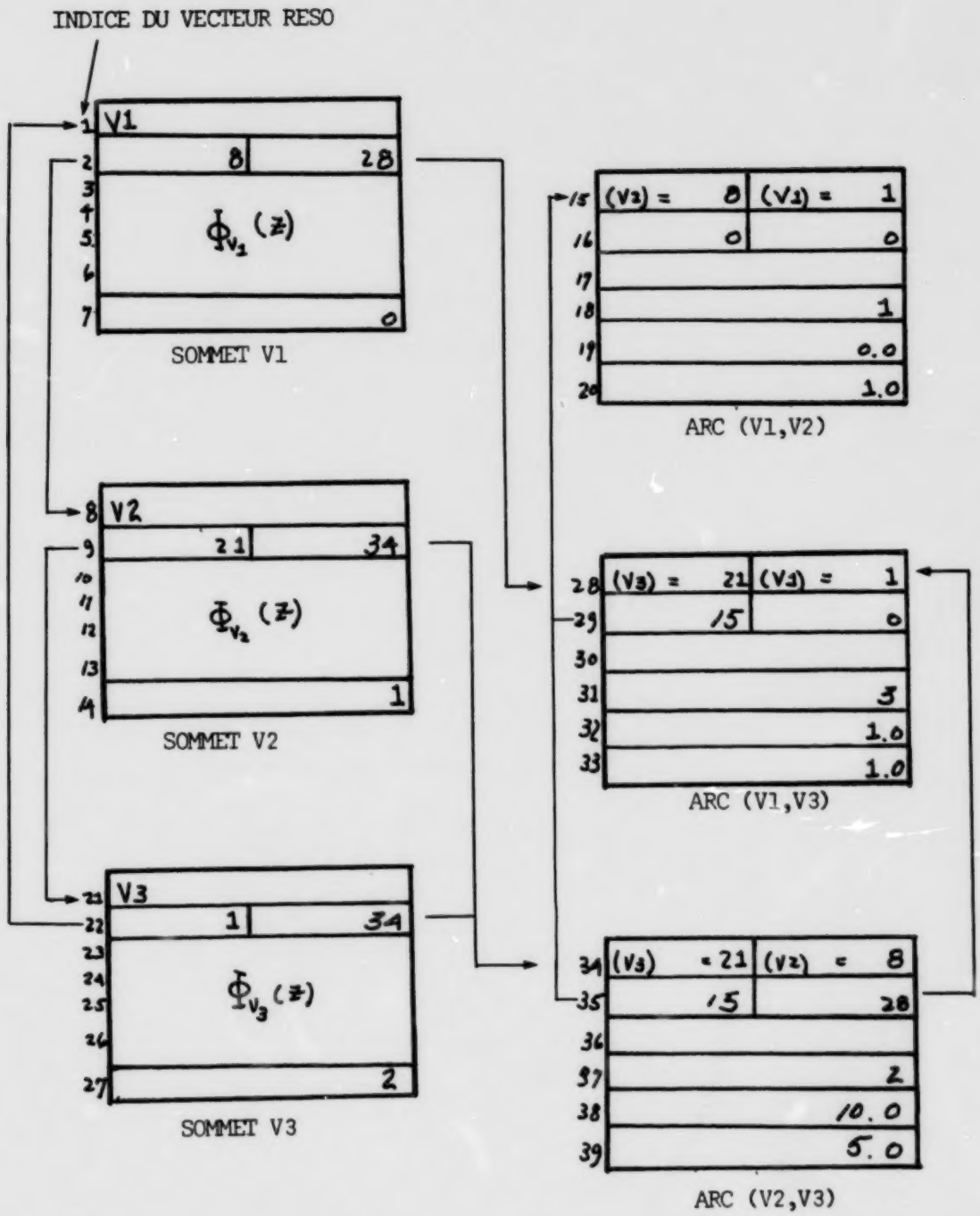


figure III.3 Contenu du vecteur RESO pour l'exemple de la figure III.2 avec K = 1 .

b) Algorithme pour calculer les $\Phi_F(Z)$

Soit F le sommet pour lequel on veut connaître la fonction de distribution F_F associée au temps nécessaire pour atteindre ce sommet. La fonction $\Phi_F(Z)$ égale à $\sum_{P_F} \prod_{E} \psi_e(Z)^{\delta_{ep}}$ doit être évaluée pour (K + 1) valeurs de Z . Nous allons décrire un algorithme permettant l'évaluation de ces (K + 1) valeurs au sommet F en parcourant une fois le réseau PERT ou une partie du réseau PERT.

Puisque le réseau PERT forme un graphe orienté sans circuit, on peut exprimer $\Phi_v(Z)$ à l'aide de la formule récursive:

$$\Phi_v(Z) = \sum_{E_v} \Phi_{n_1(e)}(Z)\, \psi_e(Z) \quad \text{pour tout } v \text{ dans } V \qquad \text{(III.1)}$$

où $E_v = \{e \text{ dans } E \mid n_2(e) = v\}$.

Cette formule permet de calculer de façon récursive les $\Phi_F(Z)$. En fait, les (K + 1) valeurs de $\Phi_F(Z)$ sont évaluées en parcourant les sommets du réseau PERT suivant un ordre topologique de telle sorte que le sommet v_i soit parcouru avant le sommet v_j si (v_i, v_j) appartient à E . L'algorithme progresse en calculant (K + 1) valeurs à chaque sommet qu'il rencontre et se termine lorsque le sommet F est atteint. Il suffit de balayer une fois le réseau PERT ou une partie du réseau PERT.

Les différentes étapes de l'algorithme sont les suivantes:

ETAPE 1 Trouver l'origine S du réseau PERT; ceci s'accomplit en trouvant le sommet n'ayant aucun arc entrant. Soit X ce sommet;

ETAPE 2 Affecter aux (K + 1) points de Φ_X la valeur complexe (1,0) ;

ETAPE 3 Etiqueter le sommet X , c'est-à-dire que la variable ARCENT de ce sommet prend la valeur -1 ;

ETAPE 4 Vérifier si X est le sommet F recherché; si oui c'est la fin du balayage, sinon il faut poursuivre avec l'ETAPE 5 ;

ETAPE 5 Réduire de 1 la variable ARCENT pour les sommets v tel que (X,v) appartient à E ;

ETAPE 6 Trouver un des sommets dont la variable ARCENT a la valeur 0 . Soit X ce sommet;

ETAPE 7 Affecter aux (K + 1) points de Φ_X les valeurs complexes obtenues avec l'équation (III.1) ;

ETAPE 8 Retourner à l'ETAPE 3 .

Il est évident que cet algorithme se généralise dans les cas où l'on désire étudier plusieurs fonctions de distribution lors d'un seul balayage. Il suffit de s'arrêter lorsque tous les sommets intéressants ont été balayés. Dans de tels cas, l'ETAPE 4 devient:

ETAPE 4 Vérifier si tous les sommets intéressants ont été balayés; si oui c'est la fin du balayage, sinon il faut poursuivre avec l'ETAPE 5 .

III.3 Structure de REPERT

Outre les sous-routines globales, le programme REPERT se divise en trois parties indépendantes: la lecture et la création du réseau PERT, l'impression du réseau PERT et l'évaluation des différentes bornes relatives aux fonctions de distribution. Chacune de ces parties exige un ensemble de sous-routines, implique un bloc de données sur le fichier d'entrée et déclenche une série d'impressions. Ces parties sont mises en exécution par des mots-contrôle apparaissant au début des blocs de données. Ces mots-contrôle sont respectivement *LIRERESO*, *IMPRIMRESO* et *PERT*.

Le programme REPERT débute en donnant le contrôle à la sous-routine CONTROL qui a pour tâche de reconnaître les mots-contrôle et d'orienter l'action. Avant d'examiner en détail chacune des trois parties, nous allons définir les variables ainsi que les sous-routines globales de REPERT.

Liste des variables

MOTCTL	dernier mot-contrôle lu
RESO	vecteur entier contenant le réseau PERT sous forme de blocs-sommets et de blocs-arcs
ZESO	vecteur réel équivalent à RESO
LGRESO	dimension du vecteur RESO
IPTRESO	pointeur sur le premier mot libre du vecteur RESO
LGARC	longueur des blocs-arcs
LGND	longueur des blocs-sommets
IDEST	vecteur entier contenant le nom des sommets pour lesquels on veut étudier la fonction de districution

LGDEST	dimension du vecteur IDEST
NBDEST	nombre de sommets considérés ($\leq$ LGDEST)
NBCH	vecteur contenant le nombre de chemins entre l'origine et les différents sommets étudiés
NBW	variable équivalente à la variable K définie lors de l'introduction de l'approximation de Davies
NBWM	deux fois le nombre maximal de points pour lesquels la fonction ϕ_F est évaluée ($2\ (NBW + 1) \leq NBWM$)
NBW1	nombre de points pour lesquels la fonction ϕ_F est évaluée ($NBW1 = NBW + 1$)
Z	vecteur réel contenant les NBW1 valeurs pour lesquelles la fonction ϕ_F est évaluée
LGZ	dimension du vecteur Z ($NBW1 \leq LGZ$)
NINIT	sommet initial d'un arc
NFINAL	sommet terminal d'un arc
IFCTDIS	type de la fonction de distribution (ex. 1 = uniforme, 2 = bêta, 3 = normale tronquée, ---)
PARA1	premier paramètre de la fonction de distribution
PARA2	second paramètre de la fonction de distribution
NBARCS	nombre d'arcs dans le réseau PERT
NBNDS	nombre de sommets dans le réseau PERT
IOR	entrée dans le vecteur RESO où débute le bloc-sommet qui correspond à l'origine du réseau PERT
MOYENNE	variable égale à 1 si on veut les bornes sur les différents moments sinon égale à 0
DELTA	variable servant dans l'approximation de Davies (Δ)

A — premier point sur l'axe des temps où F_F est évaluée. Il faut que $F_F(A) = 0$

B — dernier point sur l'axe des temps où F_F est évaluée. Il faut que $F_F(B) = 1$

C — incrément qui sert à déterminer les valeurs intermédiaires entre A et B pour lesquelles la fonction F_F est évaluée, soit aux points (A, A + C , A + 2 C , ---, B)

NBPAS — nombre de points pour lesquels les fonctions à intégrer (si nécessaire) vont être évaluées par unité de temps

IDEMI — constante octale (7777777777B) qui permet l'accès aux demi-mots.

Les sous-routines globales sont:

REPERT — qui initialise le fichier d'impression et donne le contrôle à la sous-routine CONTROL ;

SECPU — qui imprime le temps d'utilisation de l'unité centrale lors de l'appel de cette sous-routine (CPU Time) ;

ERREUR (NO) — qui imprime un message d'erreur ou d'information selon la valeur du paramètre NO ;

CONTROL — qui lit les mots-contrôle et oriente l'action.

De plus, il y a diverses fonctions complexes:

PHI (Z, I, PARA1, PARA2) qui rend la valeur de la fonction caractéristique de type I, avec paramètres PARA1 et PARA2, évaluée en Z (cette fonction doit être complétée par l'usager s'il y a plus de quatre types de fonctions de distribution) ;

FCTI (Z, PARA1, PARA2) qui évalue une fonction caractéristique de paramètres PARA1, PARA2 en Z (l'usager doit fournir autant de ces fonctions qu'il y a de types de fonctions de distribution).

Nous allons présenter les trois parties de REPERT commandées par les différents mots-contrôle. Ceci nous permettra de connaître les ENTREES / SORTIES de REPERT.

a) mot-contrôle: *LIRERESO*

Le mot-contrôle *LIRERESO* apparaissant dans les huit premières colonnes de la première carte de ce bloc de données déclenche la lecture et la création du réseau PERT. En fait, la sous-routine LIRE se charge de cette tâche. L'entrée du réseau PERT s'accomplit en décrivant sur les cartes de données qui suivent le mot-contrôle les arcs du réseau PERT. Un arc est représenté par une carte répondant à l'énoncé FORTRAN FORMAT (2A10, 1X, I1, F5.1, F5,1) . La carte comprend six champs:

CHAMP 1 :	colonnes 1-10	le nom du sommet d'origine de l'arc, NINIT;
CHAMP 2 :	colonnes 11-20	la destination de l'arc, NFINAL ;
CHAMP 3 :	colonne 21	un blanc ;
CHAMP 4 :	colonne 22	un nombre de 1 à 9 indiquant le type de la fonction de distribution associée au temps pour accomplir cette activité, IFCTDIS ;
CHAMP 5 :	colonnes 23-27	la valeur du premier paramètre de la fonction de distribution, PARA1 ;
CHAMP 6 :	colonnes 28-32	la valeur du second paramètre de la fonction de distribution, PARA2 .

La lecture s'arrête sur une carte contenant le mot FIN dans les colonnes 1, 2 et 3. La carte FIN produit l'impression du nombre d'arcs et du nombre de sommets dans le réseau PERT créé.

b) mot-contrôle: *IMPRIMRESO*

Le mot-contrôle *IMPRIMRESO* apparaissant dans les dix premières colonnes de la première carte de ce bloc de données déclenche l'impression du réseau PERT. La sous-routine IMPRIME accomplit ce travail. Ce bloc de données ne contient pas d'autres données. L'impression du réseau PERT se fait sous forme de tableau. L'exemple présenté plus loin illustre un tel tableau.

c) mot-contrôle: *PERT*

La troisième partie est déclenchée par le mot-contrôle *PERT* apparaissant dans les quatre premières colonnes de la première carte de ce bloc de données. Cette partie comprend trois sous-routines: DISTRIB, TROUVOR(IOR) et AFFECTE(IOR). DISTRIB lit les données appartenant à ce bloc, appelle les sous-routines TROUVOR(IOR) et AFFECTE(IOR), estime les différentes fonctions de distribution et évalue, si nécessaire, les bornes supérieures aux différents moments. TROUVOR(IOR) trouve l'origine du réseau PERT et IOR contient au retour l'entrée dans le vecteur RESO du bloc-sommet associé à l'origine du réseau PERT. AFFECTE(IOR) effectue le balayage du réseau PERT afin de calculer les $\phi_v(Z)$. IOR contient lors de l'appel, l'entrée dans le vecteur RESO du bloc-sommet associé à l'origine du réseau PERT. La carte suivante en entrée comprend six champs et doit répondre à l'énoncé FORTRAN FORMAT (I3, 2X, F10.5, I3, 2X, I1) :

CHAMP 1 :	colonnes 1-3	un nombre qui indique la valeur prise par NBW ;
CHAMP 2 :	colonnes 4-5	deux blancs ;
CHAMP 3 :	colonnes 6-15	la valeur prise par la variable DELTA ;
CHAMP 4 :	colonnes 16-18	le nombre de sommets pour lesquels la fonction de distribution va être calculée, NBDEST ;
CHAMP 5 :	colonnes 19-20	deux blancs ;
CHAMP 6 :	colonne 21	la valeur 1 si on veut obtenir des bornes supérieures pour les deux premiers moments associés aux fonctions de distribution que nous estimons, sinon, la valeur 0 , MOYENNE .

Sur les cartes suivantes (il y en a autant qu'il le faut), on doit préciser les NBDEST sommets pour lesquels on va calculer les fonctions de distribution et le nombre de chemins de l'origine à ces sommets suivant le FORMAT (4 (A10, 5X, I5)). Il y a douze champs par carte:

CHAMP 1 :	colonnes 1-10	le nom d'un sommet pour lequel on veut étudier la fonction de distribution ;
CHAMP 2 :	colonnes 11-15	cinq blancs ;
CHAMP 3 :	colonnes 16-20	le nombre de chemins de l'origine au sommet précédemment nommé ;

et ainsi de suite.

Les fonctions de distribution sont évaluées pour différentes valeurs entre A et B; soit en A, A + C, A + 2C, ..., B . A et B sont des entiers tels que $0 < A < B$ avec $F_F(A) = 0$ et $F_F(B) = 1$ pour tout sommet F dont on étudie les bornes pour les deux premiers moments. La

carte suivante répondant au FORMAT (2I5, F10.5) précise les valeurs de A, B et C :

CHAMP 1 : colonnes 1-5 la valeur de la variable A ;

CHAMP 2 : colonnes 6-10 la valeur de la variable B ;

CHAMP 3 : colonnes 11-20 la valeur de la variable C .

Enfin, la carte suivante doit apparaître seulement si on veut obtenir des bornes pour les moments. Elle précise en FORMAT(I3)en combien de points la fonction à intégrer va être évaluée par unité de temps:

CHAMP 1 : colonnes 1-3 la valeur de la variable NBPAS .

Ceci complète le bloc de données pour cette partie. L'impression contient la valeur des variables K et DELTA , l'approximation des fonctions de distribution aux sommets demandés ainsi que les bornes pour les moments à ces sommets (si nécessaire).

De plus, les données lues sont comprises entre >< dans le fichier de sortie. Le temps d'occupation de l'unité centrale est indiqué après l'indicatif /// lorsqu'une nouvelle action débute.

Dans la version présentée à la section III.5, REPERT permet de calculer ϕ_F en un maximum de cent points:

NBWM = 200 .

Il faut que NBW < 100 .

Donc, chaque arc occupe six mots-mémoire et chaque sommet deux cent trois mots-mémoire:

LGARC = 6

LGND = 2 * NBW1 + 3 = 203 .

Lors d'un seul balayage, on peut étudier au maximum dix fonctions de distribution:

$$LGDEST = 10$$

$$NBDEST \leq 10 \quad .$$

La taille du réseau PERT que ce programme peut traiter dépend de la dimension maximale (LGRESO) que peut prendre le vecteur RESO. Il faut satisfaire la contrainte:

$$NBNDS * LGND + NBARCS * LGARC \leq LGRESO \quad .$$

Enfin, quatre types de fonctions de distribution sont possibles.

III.4 Exemple numérique

Afin d'illustrer le mode d'utilisation de REPERT, nous avons choisi de considérer un exemple simple présenté par Martin (1965) et repris au chapitre V .

Le réseau PERT apparaît à la figure III.4 .

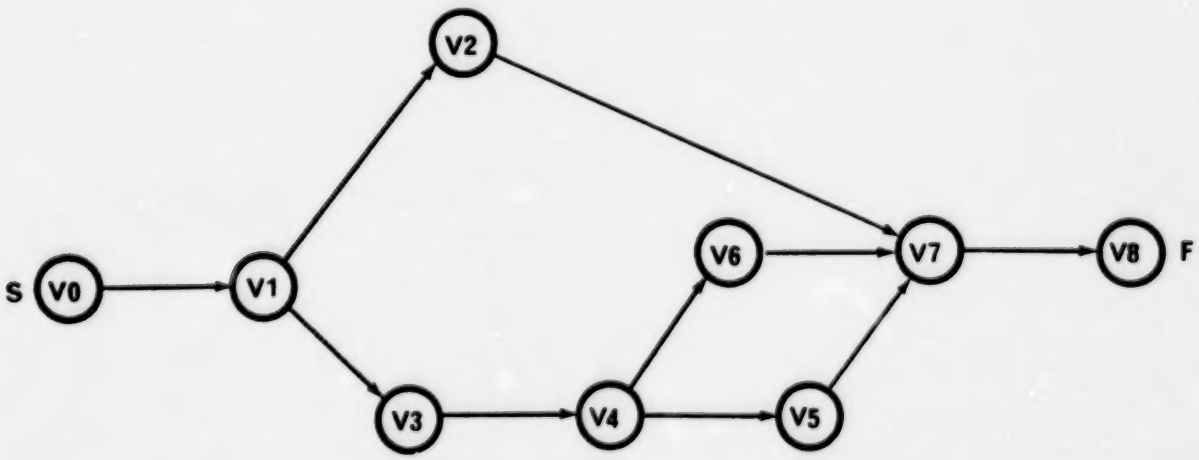


figure III.4 Exemple d'un réseau PERT proposé par Martin.

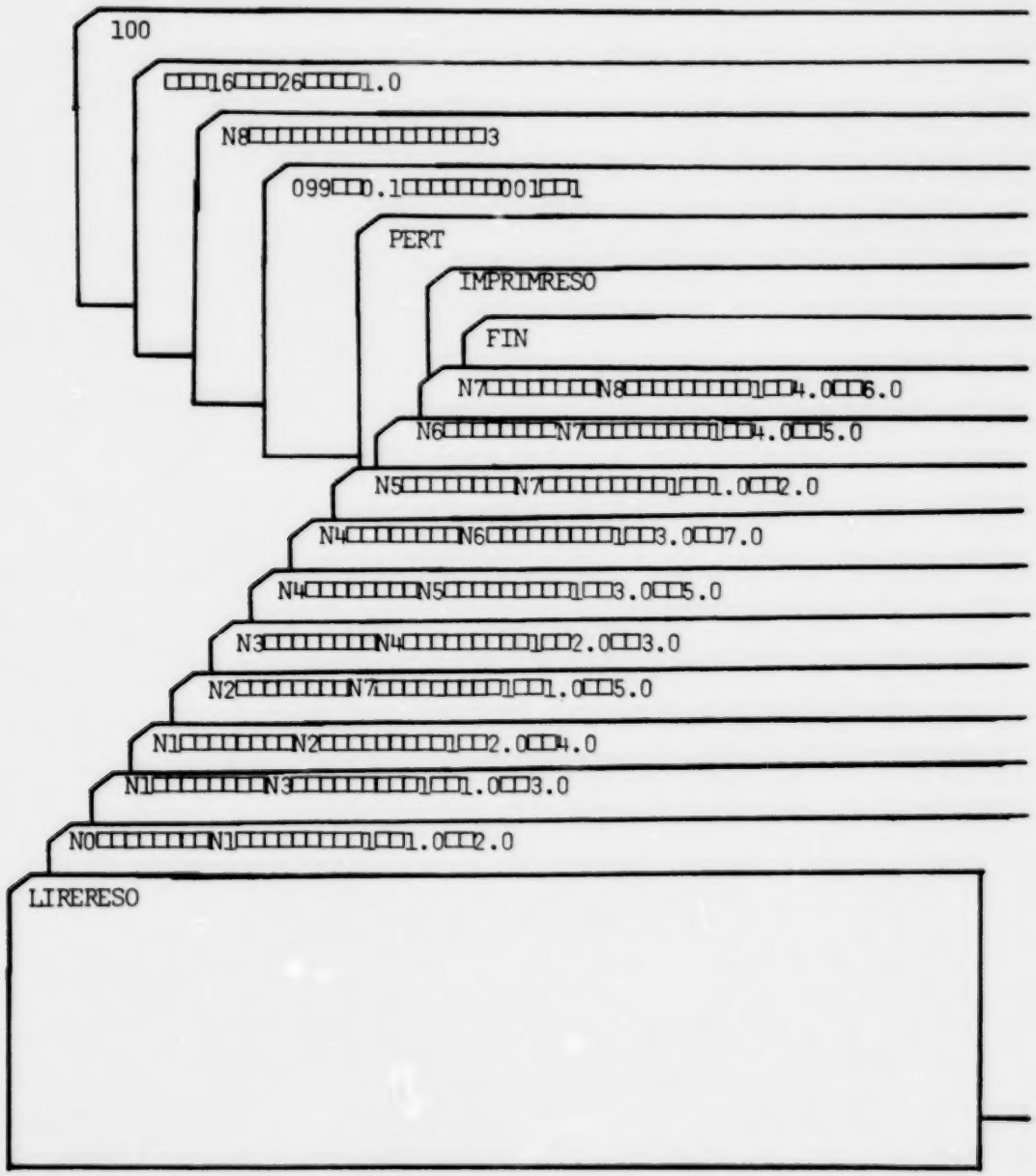
Les temps requis pour accomplir les activités suivent une loi de distribution uniforme. Si $f_{ij}(x)$ est la fonction de densité associée à l'arc (v_i,v_j), les données du problème sont alors les suivantes:

$$f_{01}(x) = 1, \qquad 1 \leq x \leq 2,$$
$$f_{12}(x) = \tfrac{1}{2}, \qquad 2 \leq x \leq 4,$$
$$f_{13}(x) = \tfrac{1}{2}, \qquad 1 \leq x \leq 3,$$
$$f_{27}(x) = \tfrac{1}{4}, \qquad 1 \leq x \leq 5,$$
$$f_{34}(x) = 1, \qquad 2 \leq x \leq 3,$$
$$f_{45}(x) = \tfrac{1}{2}, \qquad 3 \leq x \leq 5,$$
$$f_{46}(x) = \tfrac{1}{4}, \qquad 3 \leq x \leq 7,$$
$$f_{57}(x) = 1, \qquad 1 \leq x \leq 2,$$
$$f_{67}(x) = 1, \qquad 4 \leq x \leq 5,$$
$$f_{78}(x) = \tfrac{1}{2}, \qquad 4 \leq x \leq 6.$$

Nous sommes intéressés à étudier la fonction de distribution de la durée du programme au sommet v8 .

a) Fichier d'entrée (INPUT)

□ = blanc .



b) Fichier de sortie (OUTPUT)

```
///  I•372 CP SEC.

         >>LIRERESO  <<
///  I•377 CP SEC.

         >>V0        V1          1   1.0   2.0<<
         >>V1        V3          1   1.0   3.0<<
         >>V1        V2          1   2.0   4.0<<
         >>V2        V7          1   1.0   5.0<<
         >>V3        V4          1   2.0   3.0<<
         >>V4        V5          1   3.0   5.0<<
         >>V4        V6          1   3.0   7.0<<
         >>V5        V7          1   1.0   2.0<<
         >>V6        V7          1   4.0   5.0<<
         >>V7        V8          1   4.0   6.0<<
         >>FIN       <<

                    DANS LE RESEAU,IL Y A     10 ARCS ET      9 NOEUDS
         >>IMPRIMRESO<<
///  I•396 CP SEC.
```

IMPRESSION DU RESEAU

NO.ARC	ND INITIAL	NC FINAL	D+(NDI)	D+(NDF)	FCTDIST	PARA1	PARA2
1	V0	V1	0	1	1	1.000	2.000
2	V1	V2	1	1	1	2.000	4.000
3	V1	V3	1	1	1	1.000	3.000
4	V3	V4	1	1	1	2.000	3.000
5	V2	V7	1	3	1	1.000	5.000
6	V7	V8	3	1	1	4.000	6.000
7	V4	V6	1	1	1	3.000	7.000
8	V4	V5	1	1	1	3.000	5.000
9	V5	V7	1	3	1	1.000	2.000
10	V6	V7	1	3	1	4.000	5.000

D+(NDI) = NOMBRE D ARCS ENTRANT AU NOEUD INITIAL
D+(NDF) = NOMBRE D ARCS ENTRANT AU NOEUD FINAL

```
        >>PERT      <<
///  1.421 CP SEC.

        >> 99       .10000  1  1<<
        >>VB                   3
        >>   16   26   1.00000<<
              K =  99  DELTA =   .1000000



        D I S T R I B U T I O N  A U  M O Y E N      VB
        PR( T <  16.00 ) = 0.00000
        PR( T <  17.00 ) = 0.00000
        PR( T <  18.00 ) = 0.00000
        PR( T <  19.00 ) =  .16111
        PR( T <  20.00 ) =  .38003
        PR( T <  21.00 ) =  .61979
        PR( T <  22.00 ) =  .82830
        PR( T <  23.00 ) =  .95330
        PR( T <  24.00 ) =  .99470
        PR( T <  25.00 ) =  .99991
        PR( T <  26.00 ) = 1.00000
///  1.727 CP SEC.

        >>100<<

        AVEC UN PAS D INTEGRATION =   .010
        LA BORNE  SUPERIEURE POUR LA MOYENNE =   20.56227
        LA BORNE SUPERIEURE POUR LE DEUXIEME MOMENT =     424.759173
///  9.181 CP SEC.
```

III.5 Enoncés du programme REPERT

```
PROGRAM REPERT     74/74   CPT=1                    FTN 4.0+P353      11
      PROGRAM REPERT
     1(OUTPUT,INPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      C
      C X X X
```

```
C
C     INPUT/OUTPUT:
C
C     L INPUT SE FAIT PAR L ENTREMISE DU FICHIER   INPUT.
C     L OUTPUT CORRESPOND A L IMPRIMANTE (CE QUI PEUT ETRE MODIFIE PAR
C     L ENONCE     PROGRAM  ).
C     L INPUT  EST COMPOSE DE DIFFERENTS BLOCS INITIALISES PAR UN MOT-DE-
C     CONTROLE.CHAQUE MOT-DE-CONTROLE EXIGE UNE SUITE DE DONNEES ET ENGEN
C     DRE UN OUTPUT.REGARDONS DE PLUS PRES CHACUN DES BLOCS.
C
C     LIRERESC
C     --------
C          CE MOT-DE-CONTROLE APPARAISSANT DANS LES 8  PREMIERES COLONNES
C          DE LA PREMIERE CARTE DE CE BLOC DECLENCHE LA LECTURE DU RESEAU.
C          L ENTREE DU RESEAU S ACCOMPLIT EN DECRIVANT SUR LES CARTES QUI
C          SUIVENT LES ARCS DU RESEAU.UN ARC  EST REPRESENTE  PAR L IMAGE
C          D UNE CARTE REPONDANT A L ENONCE  FORTRAN    FORMAT(2A10,1X,I1
C          ,F5.1,F5.1).LA CARTE COMPREND SIX CHAMPS:
C
C          CHAMP 1 : COLONNES 1-10 .C EST LE NOM DU NOEUD D ORIGINE DE
C                                  L ARC.NINIT
C
C          CHAMP 2 : COLONNES 11-20.C EST LA DESTINATION DE L ARC.NFINAL
C
C          CHAMP 3 : COLONNE  21   .C EST UN BLANC.
C
C          CHAMP 4 : COLONNE  22   .COMPREND UN NOMBRE  DE 1 A 9  INDIQUA
C                                   NT LE TYPE DE LA FONTION DE DISTRIBUT
C                                   ION ASSOCIEE AU TEMPS POUR ACCOMPLIR
C                                   CETTE ACTIVITE.IFCTDIS
C
C          CHAMP 5 : COLONNE  23-27.UN NOMBRE INDIQUANT  LA VALEUR DU PRE
C                                   MIER PARAMETRE  DE LA FONTION DE DIST
C                                   RIBUTION ASSOCIEE A CET ARC.PARA1
C          CHAMP 6 : COLONNE  28-32.UN NOMBRE INDIQUANT  LA VALEUR DU DEU
C                                   XIEME PARAMETRE DE LA FONTION DE DIST
C                                   RIBUTION ASSOCIEE A CET ARC.PARA2
C
C          LA LECTURE S ARRETE SUR UNE CARTE CONTENANT LE MOT  FIN
C          DANS LES COLONNES  1, 2 ET 3.
C          LA CARTE  FIN   PRODUIT  L IMPRESSION DU NOMBRE D ARCS ET DU
C          NOMBRE DE SOMMETS DANS LE RESEAU LU.
C
C     IMPRIMRESO
C     ----------
C          CE MOT-DE-CONTROLE APPARAISSANT DANS LES  10 PREMIERES COLONNE
C          S DE LA PREMIERE CARTE DE CE BLOC DECLENCHE L IMPRESSION DU RE
C          RESEAU.L IMPRESSION DU RESEAU  SE FAIT SOUS FORME DE TABLEAU.
C
C     PERT
C     ----
C          CE MOT-DE-CONTROLE APPARAISSANT DANS LES 4  PREMIERES COLONNES
C          DE LA PREMIERE CARTE DE CE BLOC DECLENCHE L EVALUATION DES FON
C          CTIONS DE DISTRIBUTION ET  DES BORNES SUIVANT LES PRECISIONS
C          FOURNIENT PAR  LES CARTES DE DONNEES OBLIGATOIRES QUI SUIVENT.
C          LA CARTE SUIVANTE COMPREND SIX CHAMPS  REPONDANT A L ENONCE
C          FORTRAN   FORMAT(I3,2X,F10.5,I3,2X,I1);
```

PROGRAM REPFRT 74/74 CPT=1 FTN 4.0+P353 11.

```
C
C          CHAMP 1 : COLONNES  1-3   .CE NOMBRE INDIQUE LA VALEUR PRISE
C                                     PAR LA VARIABLE  NBW DEFINIE DANS
C                                     LE RAPPORT TECHNIQUE.NBW
C
C          CHAMP 2 : COLONNES 4-5    .DES BLANCS
C
C          CHAMP 3 : COLONNES 6-15   .CE NOMBRE EST LA VALEUR PRISE PAR LA
C                                     VARIABLE   DELTA   DEFINIE  DANS
C                                     LE RAPPORT TECHNIQUE.DELTA
C
C          CHAMP 4 : COLONNES 16-18  .CE NOMBRE INDIQUE LE NOMBRE DE NOEUDS
C                                     POUR LESQUELS LA FONCTION DE DISTRIB
C                                     UTION VA ETRE CALCULEE.NBDEST
C
C          CHAMP 5 : COLONNES 19-20  .DES BLANCS
C
C          CHAMP 6 : COLONNE  21     .CE NOMBRE EGALE 1 SI ON VEUT OBTENIR
C                                     DES BORNES SUPERIEURES SUR LES DEUX
C                                     PREMIERS MOMENTS ASSOCIES AUX FONCTIO
C                                     NS DE DISTRIBUTION QU ON CALCULE,SINO
C                                     N,CE NOMBRE DOIT EGALER 0,MOYENNE
C
C          SUR  LES  CARTES SUIVANTES(IL Y EN A AUTANT QU IL LE FAUT),IL
C          FAUT PRECISER  LES   NBDEST  SOMMETS POUR LESQUELS ON VA
C          CALCULER LA FONCTION DE DISTRIBUTION ET LE NOMBRE DE CHEMINS
C          DE L ORIGINE  DU RESEAU  PERT A CES SOMMETS:SUIVANT LE FORMAT
C          (4(A10,5X,I5)).IL Y A 12 CHAMPS PAR CARTE:
C
C          CHAMP 1 : COLONNES  1-10  .C EST LE NOM D UN SOMMET POUR LE-
C                                     QUEL ON VEUT CALCULER LA FONCTION
C                                     DE DISTRIBUTION.
C
C          CHAMP 2 : COLONNES 11-15  .DES BLANCS.
C
C          CHAMP 3 : COLONNES 16-20  .LE NOMBRE DE CHEMINS DE L ORIGINE
C                                     AU SOMMET PRECEDEMMENT NOMME.
C          ET AINSI DE SUITE.
C
C          LES FONCTIONS DE DISTRIBUTION SONT EVALUEES POUR DIFFERENTES
C          VALEURS ENTRE  A  ET  B;SOIT EN A,A+C,A+2C,....B.A ET  B SONT
C          DES ENTIERS TELS QUE:0<A<B.LA CARTE SUIVANTE REPONDANT AU
C          FORMAT(2I5,F10.5) PRECISE LES VALEURS DE  A,  B ET C:
C
C          CHAMP 1 : COLONNES  1-5   .C EST LA VALEUR DE LA VARIABLE A.
C
C          CHAMP 2 : COLONNES  6-10  .C EST LA VALEUR DE LA VARIABLE B.
C
C          CHAMP 3 : COLONNES 11-20  .C EST LA VALEUR DE LA VARIABLE C.
C
C          ENFIN,LA CARTE SUIVANTE DOIT APPARAITRE SEULEMENT SI ON VEUT
C          LES  BORNES SUR LES MOMENTS.ELLE PRECISE EN FORMAT(I3) EN
C          COMBIEN DE POINTS LA FONTION A INTEGRER VA ETRE EVALUEE PAR
C          UNITE DE TEMPS:NBPAS.
C
C          CHAMP 1 : COLONNES  1-3   .C EST LA VALEUR DE LA VARIABLE NBPAS.
```

PROGRAM REPERT 74/74 CPT=1 FTN 4.0+P353 11

```
C
C          CECI COMPLETE L INPUT.L OUTPUT INDIQUE LES VALEURS DES VARIA
C          BLES  K ET  DELTA,LES FONCTIONS DE DISTRIBUTION AINSI QUE LES
C          BORNES AUX SOMMETS DEMANDES.
C
C          DE PLUS,L OUTPUT COMPREND ENTRE LES  DELIMITEURS  >>   << LES
C          DONNEES LUES.LE TEMPS  CPU  LORSQU UNE  NOUVELLE PARTIE DU
C          PROGRAMME DEBUTE EST INDIQUE APRES L INDICATIF ///.
C
C
C     REFERENCE:
C          POUR PLUS DE DETAILS,IL FAUT SE REFERER AU RAPPORT TECHNIQUE
C          SUIVANT:
C
C          TRAHAN,MICHEL, MANUEL D UTILISATION DU PROGRAMME REPERT:RESEAU
C          ,RAPPORT TECHNIQUE NO.47,DEPARTEMENT D INFORMATIQUE,
C          UNIVERSITE DE MONTREAL,JANVIER 1974.
C
C
C     REMARQUE:
C          L UTILISATEUR DOIT FOURNIR AUTANT DE FONCTIONS CARACTERISTIQU
C          ES QU IL Y A  DE TYPES D ARCS.
C X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X X
C
C
C
C         PROGRAMME PRINCIPALE:INITIALISATION DU FICHIER OUTPUT  ET
C                              APPELLE LA SOUS-ROUTINE   CONTROL.
C
      PRINT 500
      CALL SECPU
      CALL CONTROL
      STOP 1
  500 FORMAT(1H1)
      END
```

BLOCK DATA 74/74 OPT=1 FTN 4.0+P353 11

```
      BLOCK DATA
C
C        INITIALISATION DES DIFFERENTES VARIABLES.
C
      COMMON /CTE1/ IDEMI
      COMMON /RESO/ RESO(10000),LGRESO,IPTRESO
      COMMON /CTE2/ LGARC,LGND,NBWM,NBW,MOYENNE
      COMMON /AFF / IDEST(10),LGDEST,NBDEST,Z(100),LGZ
      INTEGER RESO
      DATA IDEMI /7777777777B/
      DATA LGARC,LGND,NBWM,LGRESO,LGDEST,LGZ /6,203,200,10000,10,100/
      END
```

```
      SUBROUTINE  SECPU
C
C        SOUS-ROUTINE  QUI IMPRIME LE TEMPS CPU.
C
      CALL SECOND(T)
      WRITE(6,500) T
      RETURN
  500 FORMAT(4H ///,F7.3,* CP SEC.*,/)
      END
```

SUBROUTINE ERRFUR 74/74 CPT=1 FTN 4.0+P353 11

```
      SUBROUTINE ERREUR(NO)
C
C        IMPRIME UN MESSAGE D ERREUR  OU  D INFORMATION.
C        PARAMETRE:
C             NO = NUMERO D ERREUR
C
      GOTO (1,2,3,4,5,6,7,8,9),NO
    1 WRITE(6,501)
      CALL EXIT
    2 WRITE(6,502)
      CALL EXIT
    3 WRITE(6,503)
      CALL EXIT
    4 WRITE(6,504)
      CALL EXIT
    5 WRITE(6,505)
      CALL EXIT
    6 WRITE(6,506)
      CALL EXIT
    7 WRITE(6,507)
      CALL EXIT
    8 WRITE(6,508)
      CALL EXIT
    9 WRITE(6,509)
      CALL EXIT
  501 FORMAT(* /// MOT-DE-CONTROLE  INCONNU OU INATTENDU.   --- EXIT*)
  502 FORMAT(* /// IL Y A TROP DE DESTINATIONS  VOIR VECTEUR  IDEST .
     1--- STOP*)
  503 FORMAT(* /// IL MANQUE LA CARTE  FIN  DANS LA DESCRIPTION DU RESEA
     1U.   --- EXIT*)
  504 FORMAT(* /// MANQUE D ESPACE DANS LE VECTEUR  RESO.   --- EXIT*)
  505 FORMAT(* /// IL N Y A  PAS  D ORIGINE  DANS LE RESEAU.  --- EXIT*)
  506 FORMAT(* /// IL Y A TROP D ORIGINES DANS LE RESEAU.   --- EXIT*)
  507 FORMAT(* /// ERREUR DANS LE RESEAU:IMPOSSIBLE D EFFECTUER L AFFECT
     1ATION.  ---EXIT*)
  508 FORMAT(*  ERREUR DANS LE TYPE DE FONCTION SUR UN ARC. ---EXIT*)
  509 FORMAT(*   DEMANDE TROP DE PRECISION (NBW TROP GRAND). --- EXIT*)
      END
```

SUBROUTINE CONTROL 74/74 CPT=1 FTN 4.0+P353 11

```
      SUBROUTINE CONTROL
C
C        SOUS-ROUTINE DE CONTROLE QUI LIT LES PRINCIPALES COMMANDES
C                                 ET  ORIENTE L ACTION.
C
    1 READ(5,500) MOTCTL
      IF(EOF(5))  2,3
    2 RETURN
    3 WRITE(6,501) MOTCTL
      IF( MOTCTL .NE. 10HLIRERESO   ) GOTO  4
      CALL LIRE
      GOTO 1
    4 IF( MOTCTL  .NE. 10HIMPRIMRESO) GOTO 5
      CALL IMPRIME
      GOTO 1
    5 IF(MOTCTL .NE. 10HFERT        )GO TO 99
      CALL DISTRIB
      GO TO 1
   99 CALL ERREUR(1)
  500 FORMAT(A10)
  501 FORMAT(  10X,*>>*,A10,*<<*)
      END
```

```
      SUBROUTINE LIRE
C
C        LECTURE ET CREATION DU RESEAU
C
      COMMON /CTE1/ IDEMI
      COMMON /CTE2/ LGARC,LGND,NBWM,NBW,MOYENNE
      COMMON /RESO/ RESO(10000),LGRESO,IPTRESO
      DIMENSION ZESO(10000)
      INTEGER RESO
      EQUIVALENCE(RESO,ZESO)
      DATA IPTRESO /1/
      CALL SECPU
C
C        LECTURE D UN ARC.
C
    1 READ(5,500) NINIT,NFINAL,IFCTDIS,PARA1,PARA2
      IF(EOF(5)) 2,3
    2 CALL ERREUR(3)
    3 IF(NINIT .EQ.10HFIN          ) GOTO 900
      WRITE(6,501)NINIT,NFINAL,IFCTDIS,PARA1,PARA2
      IF(IPTRESO .NE.1) GOTO 10
C
C        RIEN N EXISTE
C        CREATION DE DEUX NOEUDS.
C
      IPTRESO= 2*LGND + 1
      RESO(1)=NINIT
      INTER=LGND+1
      RESO(2)= SHIFT(INTER,30) + IPTRESO
      RESO(3+NBWM)=0
      RESO(INTER)=NFINAL
      RESO(INTER+1) = SHIFT(1,30)+IPTRESO
      RESO(INTER+2+NBWM)=0
      NBNDS=2
      NBARCS=0
      IADR=1
      JADR=INTER
      ISORT=IENTRE=0
C
C        LES NOEUDS EXISTENT
C        CREATION D UN ARC
C
    6 RESO(IPTRESO)= SHIFT(JADR,30)+ IADR
      RESO(IPTRESO+1)=SHIFT(ISORT,30)+ IENTRE
      RESO(IPTRESO+3)=IFCTDIS
      ZESO(IPTRESO+4)=PARA1
      ZESO(IPTRESO+5)=PARA2
      RESO(JADR+2+NBWM)=RESO(JADR+2+NBWM)+1
      IPTRESO=IPTRESO+LGARC
      IF(IPTRESO .GT.LGRESO) CALL ERREUR(4)
      NBARCS=NBARCS +1
      GOTO 1
C
C        ON CHERCHE LES NOEUDS NINIT ET  NFINAL.
C
   10 IADR=JADR=0
      NOEUD=1
```

```
UBROUTINE LIRF       74/74   OPT=1                          FTN 4.0+P353      11

      12 IF(RESO(NOEUD).NE.NINIT) GOTO 15
         IADR=NOEUD
         ISORT=RESO(NOEUD+1).AND.IDEMI
         IF(JADR.EQ.0) GOTO 20
      14 RESO(IADR+1)= (RESO(IADR+1).AND. .NOT. IDEMI) +IPTRESO
         RESO(JADR+1)=(RESO(JADR+1).AND. .NOT. IDEMI) +IPTRESO
         GOTO 6
      15 IF(RESO(NOEUD).NE.NFINAL) GOTO20
         JADR=NOEUD
         IENTRE=RESO(NOEUD+1).AND.IDEMI
         IF(IADR.NE.0) GOTO 14
      20 KADR=NOEUD
         NOEUD=SHIFT(RESO(NOEUD+1),30).AND.IDEMI
         IF(NOEUD.NE.1) GOTO 12
C
C           IL FAUT CREER LES NOEUDS MANQUANT
C
         IF(IADR.NE.0) GOTO 30
         RESO(IPTRESO)=NINIT
         IND1=IPTRESO
         RESO(KADR+1)=SHIFT(IPTRESO,30)+(RESO(KADR+1).AND.IDEMI)
         RESO(IPTRESO+1) = SHIFT(1,30)
         RESO(IPTRESO+NBWD+2) = 0
         KADR=IPTRESO
         IPTRESO= IPTRESO +LGND
         IF(IPTRESO .GT.LGRESO) CALL ERREUR(4)
         NBNDS=NBNDS+1
      30 IF(JADR .NE.0) GOTO 40
         RESO(IPTRESO)= NFINAL
         IND2 =IPTRESO
         RESO(KADR+1)=SHIFT(IPTRESO,30) +(RESO(KADR+1).AND. IDEMI)
         RESO(IPTRESO +1)= SHIFT(1,30)
         RESO(IPTRESO+2+NBWD) = 0
         IPTRESO=IPTRESO +LGND
         IF(IPTRESO.GT.LGRESO) CALL ERREUR(4)
         NBNDS=NBNDS+1
      40 IF(IADR .NE.0) GOTO 45
         RESO(IND1+1)=RESO(IND1+1)+IPTRESO
         ISORT=0
         IADR=IND1
         GOTO 50
      45 RESO(IADR+1) =(RESO(IADR+1) .AND..NOT.IDEMI) + IPTRESO
      50 IF(JADR .NE. 0) GOTO 55
         IENTRE=0
         RESO(IND2+1)= RESO(IND2+1) + IPTRESO
         JADR=IND2
         GOTO 6
      55 RESO(JADR+1)=(RESO(JADR+1).AND..NOT.IDEMI) + IPTRESO
         GO TO 6
C
C           LE RESEAU EST COMPLET
C
     900 WRITE(6,590) NINIT
         WRITE(6,502) NBARCS,NBNDS
         RETURN
     500 FORMAT(2A10,1X,I1,F5.1,F5.1)
     501 FORMAT(  10X,*>>*,2A10,1X,I1,F5.1,F5.1,*<<*)
```

SUBROUTINE LIRF 74/74 OPT=1 FTN 4.0+P353 11

```
      502 FORMAT(//,20X,*DANS LE RESEAU,IL Y A*,I6,* ARCS ET*I6,* NOEUDS*)
      590 FORMAT(  10X,*>>>*,A10,*<<<*)
          END
```

UBROUTINE IMPRIME 74/74 CPT=1 FTN 4.0+P353 11

```
      SUBROUTINE IMPRIME
C
C        IMPRIME LE RESEAU SOUS FORME DE TABLEAU.
C
      COMMON /RESO/ RESO(10000),LGRESO,IPTRESO
      COMMON /CTE1/ IDEMI
      COMMON /CTE2/ LGARC,LGND,NBWM,NBW,MOYENNE
      DIMENSION ZESO(10000)
      INTEGER RESO
      EQUIVALENCE(RESO,ZESO)
      CALL SECPU
      NOEUD =1
      NOARC =0
      WRITE(6,500)
      WRITE(6,501)
    1 INTER=RES (NOEUD+1),.AND.IDEMI
      NINIT=RESO(NOEUD)
    2 INTER1=SHIFT(RESO(INTER),30) .AND. IDEMI
      IF(INTER1 .EQ.NOEUD) GOTO 10
      NFINAL=RESO(INTER1)
      NOARC=NOARC+1
      NBND1= RESO(NOEUD+2+NBWM)
      NBND2= RESO(INTER1+2+NBWM)
      IFCTDIS=RESO(INTER+3)
      PARA1  =ZESO(INTER+4)
      PARA2  =ZESO(INTER+5)
      WRITE(6,502) NOARC,NINIT,NFINAL,NBND1,NBND2,IFCTDIS,PARA1,PARA2
      INTER=SHIFT(RESO(INTER+1),30) .AND. IDEMI
      GOTO  11
   10 INTER=RESO(INTER+1).AND. IDEMI
   11 IF(INTER.NE.0) GOTO 2
      NOEUD=SHIFT(RESO(NOEUD+1),30) .AND.IDEMI
      IF(NOEUD .NE. 1) GOTO 1
      WRITE(6,503)
  500 FORMAT(1H1,10X,* I M P R E S S I O N  D U  R E S E A U *)
  501 FORMAT(/,* NO.ARC  ND INITIAL  ND FINAL    D+(NDI)  D+(NDF)  FCTDI
     1ST  PARA1  PARA2 *)
  502 FORMAT(I5,5X,A10,2X,A10,3X,I3,6X,I3,5X,I3,3X,F7.3,1X,F7.3)
  503 FORMAT(/,20X,*D+(NDI) =  NOMBRE D ARCS ENTRANT AU NOEUD INITIAL*,
     1       /,20X,*D+(NDF) =  NOMBRE D ARCS ENTRANT AU NOEUD FINAL*  ,
     2 /,1H1)
      RETURN
      END
```

SUBROUTINE DISTRIB 74/74 OPT=1 FTN 4.0+P353 11/

```
      SUBROUTINE DISTRIB
C
C        SOUS-ROUTINE QUI DIRIGUE L EVALUATION DES FONCTIONS DE DISTRIBU
C        TION AINSI QUE  DES  BORNES.
C
      COMMON /CTE2/ LGARC,LGND,NBWM,NBW,MOYENNE
      COMMON /RESO/RESC(10000),LGRESC,IPTRESC
      COMMON /AFF/ IDEST(10),LGDEST,NBDEST,Z(100),LGZ
      INTEGER RESO,A,B
      DIMENSION ZESC(10000),NBCH(10)
      EQUIVALENCE(RESO,ZESO)
      CALL SECPU
C
C        LECTURE  DES DONNEES RELATIVES A CE BLOC
C
      READ(5,500) NBW,DELTA,NBDEST,MOYENNE
      WRITE(6,501) NBW,DELTA,NBDEST,MOYENNE
      READ(5,502) (IDEST(I),NBCH(I),I=1,NBDEST)
      WRITE(6,503)(IDEST(I),NBCH(I),I=1,NBDEST)
      READ(5,504) A,B,C
      WRITE(6,505) A,B,C
      WRITE(6,508) NBW,DELTA
      NBW1=NBW+1
      IF(NBW1 .GT.LGZ) CALL ERREUR(9)
      L=2*NBW1+3
      IF(L .GT.LGND) CALL ERREUR(9)
      DO 2 I=1,NBW1
    2 Z(I)=(I-0.5)*DELTA
      CALL TROUVCR(ICR)
      CALL AFFECTE(ICR)
C
C        BOUCLE  SUR LE NOMBRE DE  DESTINATIONS
C
      DO 3 I=1,NBDEST
      NBCHEM = NBCH(I)
      L=IDEST(I)
      L1=RESC(L)
      WRITE(6,506) L1
      X=A
    4 SUM=0.0
C
C        EVALUATION DE LA FONCTION DE DISTRIBUTION POUR LES DIFFERENTES
C        DESTINATIONS
C
      DO 5 J=1,NBW1
    5 SUM=SUM+(AIMAG(CMPLX(ZESO(J*2+L),ZESO(J*2+L+1)) )
     1         *CEXP(CMPLX(0.0,-(J-0.5)*DELTA*X))))  /
     1              (3.141592?*(J-0.5))
      SUM=1.0-(NBCHEM/2.0)-SUM
      IF(SUM.LT. 0.0 ) SUM=0.0
      IF(SUM.GT. 1.0 ) SUM=1.0
      WRITE(6,507) X,SUM
      X=X+C
      IF(X.LE.B) GO TO 4
    3 CONTINUE
      CALL SECPU
      IF(MOYENNE .EQ. 0 ) RETURN
```

SUBROUTINE DISTRIB 74/74 OPT=1 FTN 4.[illegible]+P353 11

```
C
C        EVALUATION DES BORNES POUR LES DIFFERENTS MOMENTS
C
C
      READ(5,514) NBPAS
      WRITE(6,515) NBPAS
      Z1=NBPAS
      C1=1.0/Z1
      CC=0.5*C1
      DO 13 I=1,NBDEST
      NBCHEM = NBCH(I)
      L=IDEST(I)
      L1=RESC(L)
      SUM2=A
      X=A
      SUM3=A*A/2
      SUM=0.0
      DO 6 J=1,NBW1
    6 SUM=SUM +(AIMAG(CMPLX(ZFSO(J*2+1),ZESO(J*2+L+1) )
     1        *CEXP(CMPLX(0.0,-(J-0.5)*DELTA*X)))) /
     1            (3.1415927*(J-0.5))
      SUM=1.0 -(NBCHEM/2.0)-SUM
      IF(SUM .LT. 0.0) SUM=0.0
      IF(SUM .GT. 1.0) SUM=1.0
      XF=1.0-SUM
      XF3=X*XF
      DO 10 I1=A,B
      DO 10 I2=1,NBPAS
         E=I2/Z1
      X=I1+E
      SUM=0.0
      DO 7 J=1,NBW1
    7 SUM=SUM+(AIMAG(CMPLX(ZESO(J*2+L),ZFSO(J*2+L+1) )
     1        *CEXP(CMPLX(0.0,-(J-0.5)*DELTA*X)))) /
     1             (3.1415927*(J-0.5) )
      SUM=1.0 - (NBCHEM/2.0)-SUM
      IF(SUM .LT. 0.0) SUM=0.0
      IF(SUM .GT. 1.0) SUM=1.0
      YF=1.0-SUM
      YF3=X*YF
      SUM3=SUM3+CC*(XF3+YF3)
      SUM2=SUM2+CC*(XF+YF)
      XF3=YF3
   10 XF=YF
      WRITE(6,511) C1,SUM2
      SUM3=2*SUM3
      WRITE(6,513) SUM3
   13 CONTINUE
      CALL SECPU
      RETURN
  500 FORMAT(I3,2X,F10.5,I3,2X,I1)
  501 FORMAT(  10X,*>>*,I3, X,F10.5,I3,2X,I1,*<<*)
  502 FORMAT(4(A10,5X,I5))
  503 FORMAT(10X,*>>*,4(A10,5X,I5),*<<*)
  504 FORMAT(2I5,F10.5)
  505 FORMAT(  10X,*>>*,2I5,F10.5,*<<*)
  506 FORMAT(////,10X,*D I S T R I B U T I O N  A U  N O E U D*, 5X,
```

SUBROUTINE DISTRIB 74/74 OPT=1 FTN 4.0+P353 11.

```
     1  A10)
507 FORMAT(10X,*PR( T < *,F6.2,* ) = *,F7.5)
508 FORMAT(15X,* K = *,I3,*  DELTA = *,F10.7)
509 FORMAT(I3)
510 FORMAT(10X,*NOMBRE TOTAL DE CHEMINS = *,I5)
511 FORMAT(/,10X,*AVEC UN PAS D INTEGRATION = *,F6.3,/,10X,*LA BORNE
    1SUPERIEURE POUR LA MOYENNE = *,F10.5)
513 FORMAT(10X,*LA BORNE SUPERIEURE  POUR LE DEUXIEME MOMENT =*,F15.5)
514 FORMAT(I3)
515 FORMAT(10X,*>>>*,I3,*<<<*)
      END
```

SUBROUTINE TROUVOR 74/74 OPT=1 FTN 4.0+P353 11/

```
      SUBROUTINE TROUVOR(IOR)
C
C        SOUS-ROUTINE QUI TROUVE L ORIGINE DU RESEAU PERT.
C        CETTE ORIGINE DOIT ETRE UNIQUE.
C        PARAMETRE:
C             IOR= CONTIENT AU RETOUR L ENTREE DANS LE VECTEUR  RESO
C                  DE L ORIGINE  DU RESEAU PERT
C
      COMMON /RESO/ RESO(10000),LGRESO,IPTRESO
      COMMON /CTE1/ IDEMI
      COMMON /CTE2/ LGARC,LGND,NBWM,NBW,MOYENNE
      INTEGER RESO
      ICOMP=0
      NOEUD=1
    1 IF(RESO(NOEUD+2+NBWM) .NE. 0 ) GO TO 2
      IOR=NOEUD
      ICOMP=ICOMP+1
    2 NOEUD=SHIFT(RESO(NOEUD+1),30) .AND. IDEMI
      IF(NOEUD.NE.1) GOTO 1
      IF(ICOMP.EQ.0) CALL ERREUR(5)
      IF(ICOMP.GT.1) CALL ERREUR(6)
      RETURN
      END
```

SUBROUTINE AFFECTE 74/74 CPT=1 FTN 4.0+P353 11

```
      SUBROUTINE AFFECTE(IOR)
C
C        SOUS-ROUTINE QUI CALCULE LES  FONCTIONS CARACTERISTIQUES.
C        PARAMETRE:
C             IOR= CONTIENT LORS DE L APPEL L ENTREE DANS LE VECTEUR
C                  RESO    DE L ORIGINE DU RESEAU PERT
C
C
      COMMON /CTE1/ IDEMI
      COMMON /CTE2/ LGARC,LGND,NBWM,NBW,MOYENNE
      COMMON /RESO/ RESO(10000),LGRESO,IPTRESO
      COMMON /AFF/  IDEST(10),LGDEST,NBDEST,Z(100),LGZ
      INTEGER RESO
      DIMENSION ZESC(10000)
      COMPLEX SUM(100),PHI
      EQUIVALENCE(RESO,ZESC)
C
C        INITIALISATION
C
      NODE =IOR
      ISUMDES=0
      NBW1=NBW+1
      RESO(NODE+NBWM+2)=-1
      DO 1 I=1,NBW1
      ZESO(NODE+2*I) = 1.0
    1 ZESO(NODE+2*I+1) =0.0
    2 IDNOEUD=RESO(NODE)
C
C        BOUCLE POUR SAVOIR SI ON A ATTEINT TOUTES LES DESTINATIONS
C        DESIREES
C
      DO 3 I=1,NBDEST
      IF(IDEST(I).NE. IDNOEUD) GOTO 3
      ISUMDES =ISUMDES+1
      IDEST(I)=NODE
      IF(ISUMDES.GE. NBDEST) 99,4
    3 CONTINUE
    4 INTER= RESO(NODE+1).AND. IDEMI
    5 INTER1=SHIFT(RESO(INTER),30) .AND. IDEMI
      IF(INTER1.EQ. NODE) GOTO 6
      RESO(INTER1+2+NBWM)=RESO(INTER1+2+NBWM)-1
      INTER=SHIFT(RESO(INTER+1),30) .AND. IDEMI
      GOTO 7
    6 INTER=RESO(INTER+1).AND. IDEMI
    7 IF(INTER.NE.0) GOTO 5
      NODEINI=NODE
C
C        A LA RECHERCHE D UN SOMMET QUI PEUT ETRE MARQUE
C
    8 IF(RESO(NODE+2+NBWM) .EQ.0) GO TO 20
      NODE=SHIFT(RESO(NODE+1),30) .AND. IDEMI
      IF(NODE.NE.NODEINI) GOTO 8
      CALL ERREUR(7)
C
C        ON A UN SOMMET QUI PEUT ETRE  MARQUE
C
   20 RESO(NODE+NBWM+2)=-1
```

SUBROUTINE AFFECTE 74/74 CPT=1 FTN 4.0+P353 11

```
C
C       BOUCLE SUR LE NOMBRE DE POINTS OU ON EVALUE LA FONCTION CARACTE
C       RISTIQUE
C
      DO 21 I=1,NBW1
   21 SUM(I)=CMPLX(0.0,0.0)
      INTER=RESO(NODE+1).AND.IDEMI
   22 INTER1=RESO(INTER).AND.IDEMI
      IF(INTER1 .EQ. NODE) GOTO 25
      IFCTDIS=RESO(INTER+3)
      PARA1=ZESO(INTER+4)
      PARA2=ZESO(INTER+5)
      DO 23 I=1,NBW1
   23 SUM(I)=SUM(I)+CMPLX(ZESO(INTER1+2*I),ZESO(INTER1+2*I+1) )
     1       *  PHI(Z(I),IFCTDIS,PARA1,PARA2)
      INTER=RESO(INTER+1) .AND. IDEMI
      GOTO 26
   25 INTER=SHIFT(RESO(INTER+1),30).AND. IDEMI
   26 IF(INTER.NE.0) GOTO 22
      DO 28 I=1,NBW1
      ZESO(NODE+2*I)=REAL(SUM(I))
   28 ZESO(NODE+2*I+1) = AIMAG(SUM(I))
      GOTO 3
   99 RETURN
      END
```

FUNCTION PHI 74/74 CPT=1 FTN 4.0+P353 11

```
      COMPLEX FUNCTION PHI(Z,I,P1,P2)
C
C        RENC  LA VALEUR  DE LA FONCTION CARACTERISTIQUE  DE TYPE  I
C        AVEC PARAMETRE  P1 ET P2   EVALUEE EN  Z.
C
C        PARAMETRE:
C             Z = POINT OU ON EVALUE LA FONCTION CARACTERISTIQUE
C             I = TYPE  DE LA FONCTION CARACTERISTIQUE
C             P1= PREMIER PARAMETRE DE LA FONCTION CARACTERISTIQUE
C             P2= DEUXIEME PARAMETRE DE LA FONCTION CARACTERISTIQUE
C
      COMPLEX FCT1,FCT2,FCT3,FCT4
      IF(I.LT.1 .OR. I .GT. 4) CALL ERREUR(8)
      GO TO (1,2,3,4),I
    1 PHI=FCT1(Z,P1,P2)
      RETURN
    2 PHI=FCT2(Z,P1,P2)
      RETURN
    3 PHI=FCT3(Z,P1,P2)
      RETURN
    4 PHI=FCT4(Z,P1,P2)
      RETURN
      END
```

FUNCTION FCTĪ 74/74 CPT=ī FTN 4.0+P353 11

```
      COMPLEX FUNCTION FCT1(Z,P1,P2)
C
C        FONCTION COMPLEXE DEVANT ETRE  FOURNIE PAR L USAGER.
C        L ENONCE SUIVANT SERT D EXEMPLE POUR UNE FONCTION DE DISTRIBUTI
C        ON UNIFORME(P1,P2)
C
      FCT1=(CEXP(CMPLX(0.0,Z*P2))-CEXP(CMPLX(0.0,Z*P1)))  /
     ī          CMPLX(0.0,Z*(P2-P1))
      RETURN
      END
```

APPENDICE IV

LISTE DES PRINCIPALES VARIABLES APPARAISSANT DANS LES CHAPITRES TRAITANT DES PROBLEMES DE TRANSPORT

G graphe représentant un réseau de transport

$G = [V, E, n, d]$ ou $G = [V, E, n, d_{v_o}]$

V ensemble des sommets dans G

$V = \{v\}$

E ensemble des arcs dans G

$E = \{e\}$

E' sous-ensemble de E

$E' = \{e \text{ dans } E \mid x_e \text{ est disponible}\}$

$P_{v_i v_j}$ ensemble des chemins entre les sommets v_i et v_j

P_{v_j} ensemble des chemins entre les sommets v_o et v_j

P ensemble des chemins entre les sommets v_o et v_d

O-D ensemble des paires origine-destination dans G

$O\text{-}D = \{ (v_o, v_d) \}$

X ensemble ordonné de sommets indiquant l'ordre dans lequel les sommets sont parcourus dans l'algorithme d'affectation

$\overline{X}$ ensemble complémentaire de X

$V = X \cup \overline{X}$

β ensemble des éléments quantitatifs du réseau

Y ensemble des demandes en voyages entre les paires origine-destination

$Y = \{Y_q \mid q \text{ dans } O\text{-}D\}$

A algorithme d'affectation

p chemin dans G

v_o, O_q origines dans le réseau

v_d, D_q destinations dans le réseau

$q, (v_o, v_d), (O_q, D_q)$ paires origine - destination

$L(e)$ longueur de l'arc e

$n(e)$ fonction qui fait correspondre à chaque arc e dans E une paire ordonnée de sommets distincts dans V

$n(e) = (v_i, v_j) = [n_1(e), n_2(e)]$

$d(e)$ mesure d'utilité associée à l'arc e

$d_{v_o}(e,T)$ mesure d'utilité associée à l'arc e relativement à un temps de départ T de v_o

x_e charge observée sur l'arc e

X_e charge estimée sur l'arc e

x répartition partielle de trafic observée

$x = \{x_e \mid e \text{ dans } E'\}$

X répartition de trafic estimée

$X = \{X_e \mid e \text{ dans } E\}$

Y_q nombre de voyages entre la paire origine-destination q

$Y_{v_o v_d}$ nombre de voyages entre la paire origine-destination (v_o, v_d)

$p(v)$ longueur du plus court chemin entre les sommets v_o et v

$C(x,X)$ mesure de concordance entre x et X

$Pr[p]$ probabilité qu'un chemin p dans $P_{v_i v_j}$ soit utilisé pour un voyage entre les sommets v_i et v_j

$Pr[p \mid n]$ probabilité qu'un voyageur partant de v_o au temps n emprunte le chemin p dans P_v sachant qu'il doit se rendre en v

$Pr[e]$ probabilité que l'arc e soit utilisé pour un voyage entre les sommets v_o et v_d

$Pr[e,t]$ probabilité d'être sur l'arc e au temps t, sachant qu'on va de v_o à v_d

$Pr[e \mid n]$ probabilité qu'un usager partant de l'origine v_o au temps n emprunte l'arc e

$Pr[v]$ probabilité qu'un usager voyageant entre v_o et v_d passe par le sommet v

$Pr[v \mid n]$ probabilité qu'un usager passe par le sommet v sachant qu'il quitte v_o au temps n pour se rendre en v_d

T temps de départ d'un usager du sommet v_o

f_o fonction de densité associée à T

t_v temps d'arrivée au sommet v pour un usager qui passe par v, v différent de v_o

f_v fonction de densité associée à t_v

Φ_v fonction caractéristique associée à t_v

$t_{v|n}$ temps d'arrivée au sommet v pour un usager qui passe par v et dont le temps de départ de v_o est n, v différent de v_o

$f_{v|n}$ fonction de densité associée à $t_{v|n}$

$\Phi_{v|n}$ fonction caractéristique associée à $t_{v|n}$

$t_{e|n}$ temps requis à un usager partant de v_o au temps n pour parcourir l'arc e

$g_{e|n}$ fonction de densité associée à $t_{e|n}$

$\Phi_{e|n}$ fonction caractéristique associée à $t_{e|n}$

$V_{e|n}$ vitesse d'un véhicule sur l'arc e, s'il part de v_o au temps n

$t_{p|n}$ temps nécessaire à un usager partant de v_o au temps n pour parcourir le chemin p

$g_{p|n}$ fonction de densité associée à $t_{p|n}$

μ_v moyenne de f_v

σ^2_v variance de f_v

$\mu_{e|n}$ moyenne de $g_{e|n}$

$\sigma^2_{e|n}$ variance de $g_{e|n}$

$\mu'_{e|n}$ moyenne de $V_{e|n}$

$\sigma'^2_{e|n}$ variance de $V_{e|n}$

θ paramètre de calibration

a_{O_q} paramètre associé à l'origine O_q

b_{D_q} paramètre associé à la destination D_q

K constante de normalisation

M constante contenant le plus grand nombre pouvant être mémorisé par l'ordinateur utilisé

APPENDICE V

LISTE DES PRINCIPALES VARIABLES APPARAISSANT DANS LES CHAPITRES TRAITANT DES PROBLEMES D'ORDONNANCEMENT

G représentation graphique d'un réseau PERT ou d'un RPD

G' représentation graphique d'un RPDR associé au RPD G

V ensemble des sommets dans G

$V = \{v\}$

n nombre d'éléments dans V

V' ensemble des sommets dans G'

$V' = \{v'\}$

E ensemble des arcs dans G

$E = \{e\}$

m nombre d'éléments dans E

E' ensemble des arcs dans G'

$E' = \{e'\}$

E_v sous-ensemble de E

$E_v = \{e \text{ dans } E \mid n_2(e) = v\}$

θ sous-ensemble quelconque de E

θ_i sous-ensemble de E

$\theta_i = \{e \text{ dans } E \mid n_2(e) \leq i\}$

β_i sous-ensemble de θ_i

$\beta_i = \{e \text{ dans } \theta_i \mid n_2(e) = i\}$

$\overline{\beta}_i$ sous-ensemble complémentaire de β_i dans θ_i

$\overline{\beta}_i = \{e \text{ dans } \theta_i \mid n_2(e) < i\}$

E_i sous-ensemble quelconque de θ_i

$\bar{E}_i$ sous-ensemble complémentaire de E_i dans θ_i

$E_i \cup \bar{E}_i = \theta_i$

T_θ ensemble de variables aléatoires t_e

$T_\theta = \{t_e \mid e \text{ dans } \theta \subseteq E\}$

P_v ensemble des chemins entre S et v

r_v nombre d'éléments dans P_v

P ensemble des chemins entre S et F

r nombre d'éléments dans P

P.v ensemble regroupant un certain nombre de chemins se terminant en v pour lequel nous avons:

$$\ell_v = \max_{P.v} \{\ell_{n_1(p)} + t_p\}$$

p chemin quelconque dans G

$p_{v'}$ chemin entre S et v' dans V'

R chemin critique obtenu par la méthode CPM lorsque $E[t_e]$ remplace t_e pour tous les arcs du réseau

ϕ ensemble vide

S sommet-source dans G

F sommet-objectif dans G

s nombre de sommets-objectifs dans un RPD

$F_i (i = 1, \ldots, s)$ le $i^{\text{ème}}$ sommet-objectif dans un RPD

F' sommet-objectif fictif dans un RPD

$d^+(v)$ nombre d'arcs entrant dans le sommet v

$d^-(v)$ nombre d'arcs sortant du sommet v

ℓ_i durée du programme jusqu'au sommet i

g_i borne obtenue par la méthode CPM au sommet i

f_i borne proposée par Fulkerson au sommet i

s_i borne proposée par Elmaghraby au sommet i

m_i, n_i, n_i^1, n_i^2, n_i^3 bornes que nous proposons pour le sommet i

RPD réseau PERT "décisionnel"

RPDR réseau PERT "décisionnel" réduit

n(e) fonction qui fait correspondre à chaque arc e une paire ordonnée de sommets distincts:

$$n_{(e)} = (v_i, v_j) = [n_1(e), n_2(e)]$$

où v_i est le sommet initial de e et v_j son sommet terminal

n(p) fonction qui fait correspondre à un chemin p dans G une paire ordonnée de sommets distincts:

$$n(p) = (v_i, v_j) = [n_1(p), n_2(p)]$$

où v_i est le sommet initial de p et v_j son sommet terminal

t_e temps de réalisation de l'activité associée à l'arc e

f_e fonction de densité associée à t_e

F_e fonction de distribution associée à t_e

ψ_e fonction caractéristique associée à t_e

t_p temps de réalisation associé au chemin p

f_p fonction de distribution associée à t_p

ψ_p fonction caractéristique associée à t_p

T temps nécessaire pour atteindre l'objectif F

T' temps nécessaire pour atteindre le sommet-objectif dans un réseau PERT parallèle

f fonction de densité associée à T

F fonction de distribution associée à T

ψ fonction caractéristique associée à T

X variable aléatoire

E[X] moyenne de X

V[X] variance de X

$E_{T_\theta}[X]$ espérance mathématique de X impliquant les variables aléatoires de l'ensemble T_θ

$\underline{E[X]}$ borne inférieure pour E[X]

$\overline{E[X]}$ borne supérieure pour E[X]

$\underline{E[X^2]}$ borne inférieure pour $E[X^2]$

$\overline{E[X^2]}$ borne supérieure pour $E[X^2]$

$\underline{V[X]}$ borne inférieure pour V[X]

$\overline{V[X]}$ borne supérieure pour V[X]

Pr[v] probabilité d'atteindre le sommet v

$Pr[T \leq x \mid F_i]$ fonction de distribution de la durée d'un programme sachant qu'il se termine en F_i

${}_iP_A$ (i = 1,2) probabilité que l'éventualité i de l'événement "décisionnel" A se produise étant donné que le sommet A est atteint

$\left[\gamma^{-1}\right]_x$ fonction caractéristique inverse de γ évaluée en x

IM[NC] fonction qui rend la partie imaginaire du nombre complexe NC

$\lambda(x)$ borne inférieure pour $F(x)$ dans un réseau PERT

$\lambda^{+}(x)$ borne inférieure pour $F(x)$ dans un RPD

q nombre de chemins parallèles dans un réseau PERT parallèle

REMERCIEMENTS

Je désire exprimer ma gratitude à Monsieur le professeur Pierre Robillard d'avoir bien voulu accepter de diriger mes travaux. Son encouragement et ses suggestions ont été indispensables à l'élaboration de cette thèse.

Je tiens à remercier Monsieur le professeur Michael Florian pour ses nombreux conseils et pour son aide bienveillante tout au long de ce travail.

Je désire remercier tous mes collègues de l'équipe "FORD". J'estime qu'ils ont su créer un climat propice au travail de recherche.

Mes remerciements vont également à tous les professeurs du département d'informatique qui, par leur enseignement et leurs judicieux conseils, ont favorisé la préparation de ce travail, ainsi qu'à Monsieur Robert Dial qui nous a suggéré le problème traité au chapitre V.

Je remercie le Conseil national de recherches du Canada, le Centre de recherche sur les transports de l'Université de Montréal et le Centre de développement des transports du ministère des Transports du Canada pour les bourses et les subventions qu'ils m'ont octroyées au cours de ces dernières années.

Je suis très reconnaissant à Madame Jacqueline Gariépy pour son excellent travail de dactylographie.

Enfin, j'exprime toute ma gratitude à ma femme pour la patience dont elle a fait preuve durant l'élaboration de ce travail.

BIBLIOGRAPHIE

Battersby, A., Méthodes modernes d'ordonnancement, Dunod, 1967.

Beckmann, M.J., MacGuire, C.B. et Winsten, C.B., Studies in the Economics of Transportation, Yale University Press, New Haven, Connecticut, 1956.

Bohmann, H., A Method to Calculate the Distribution Function when the Characteristic Function is Known, BIT 10, pp. 237-242, 1970.

Bohmann, H., From Characteristic Function to Distribution Function via Fourier Analysis, BIT 12, pp. 279-283, 1972.

Brown, R.M., Expressway Route Selection and Vehicular Usage, Bulletin Highway Research Bd. 16, pp. 12-21, 1948.

Bruggeman, J.M., Network Flow Simulation for Urban Traffic Control System, Federal Highway Administration Office of Research, Report No. FH-11-7462-2.

Burt, John M. et Garman, Mark B., Monte Carlo Techniques for Stochastic PERT Network Analysis, INFOR, Vol. 9, no 3, 1971.

Campbell, Moses Earl, Assignment of Traffic to Expressways, South-Eastern Association of State Highway Officials, Proceedings, pp. 64-65, 1951.

Charnes, A. et Cooper, W. W., Multicopy Traffic Network Models, "Proceedings of the Symposium on Traffic Flow", R. Herman éditeur, 1961.

Charnes, A., Cooper, W.W. et Thompson, G.L., Critical Path Analyses via Chance Constrained and Stochastic Programming, Operations Research, Vol. 12, pp. 460-470, 1963.

Clark, C.E., The Greatest of a Finite Set of Random Variables, Operations Research, Vol. 9, pp. 145-162, 1961.

Clingen, C.T., A Modification of Fulkerson's Algorithm for Expected Duration of a PERT Project when Activities have Continuous d.f., Operations Research, Vol. 12, no 4, pp. 629-632, 1964.

Dafermos, S.C. et Sparrow, F.T., The Traffic Assignment Problem for a General Network, NBS J. Research Ser. B. 73, pp. 91-118, 1969.

Davies, R.B., Numerical Inversion of a Characteristic Function, Biometrika 60, pp. 415-418, 1973.

Department of Scientific and Industrial Research, Road Research Laboratory, Research on Road Traffic, London, Her Majesty's Stationery Office, 1965.

Dial, Robert, B., A Probabilistic Multipath Traffic Assignment Model which Obviates Path Enumeration, Transportation Research, Vol. 5, pp. 83-111, 1971.

Dial, Robert, B., Probabilistic Assignment: a Multipath Traffic Assignment Model which Obviates Path Enumeration, University Microfilms, 1971.

Draper, N.R. et Smith, H., Applied Regression Analysis, John Wiley and Sons, Inc., 1967.

DYNET: Lieberman, E., DYNET: a Dynamic Network Simulation of Urban Traffic Flow, "Proceedings of the Third Annual Simulation Symposium",1970.

Elmaghraby, Solak, E., On the Expected Duration of PERT Type Networks, Management Science, Vol. 13, no 5, pp. 299-306, 1967.

Feller, William, An Introduction to Probability Theory and its Applications, Vol. I, troisième édition, John Wiley and Sons, Inc., 1968.

Feller, William, An Introduction to Probability Theory and its Applications, Vol. II, deuxième édition, John Wiley and Sons, Inc., 1971.

Florian, Michael, On Modelling Congestion in Dial's Probabilistic Assignment Model, Publication no 124A, Département d'informatique, Université de Montréal, mars 1973.

Florian, Michael et Nguyen, Sang, A Method for Computing Network Equilibrium with Elastic Demands, Publication no 140, Département d'informatique, Université de Montréal, août 1973.

Ford, L.R. et Fulkerson, D.R., Solving the Transportation Problem, Management Science, Vol. 3, pp. 24-32, 1957.

Fulkerson, D.R., Expected Critical Path Lengths in PERT Networks, Operations Research, Vol. 10, pp. 808-817, 1962.

Gibert, A., A Method for the Traffic Assignment when Demand is Elastic , LBS-TNT-85, Transport Network Theory Unit, London Business School, London, 1968.

Greville, T.N.E., Spline Functions, Interpolation and Numerical Quadrature, extrait du volume Mathematical Method for Digital Computers , édité par Ralston, A. et Wilf, H.S., chez John Wiley and Sons, Inc., 1967.

Hartley, H.O. et Wortham, A.W., A Statistical Theory for Critical Path Analysis, Management Science, Vol. 12, no 10, pp. B469-B481, 1966.

Highway Research Board, Special Report 87, Highway Capacity Manual 1965, National Academy of Sciences National Research Council, 1965.

Homburger, W.S., Traffic Estimation-Computer Programs for Educational Purpose, deuxième édition, ITTE course note, University of California, Berkeley, 1969.

Hu, T.C., Integer Programming and Network Flows, Addison-Wesley, p. 151, 1969.

Huber, M.S., Boutwell, H.B. et Witheford, D.K., Comparative Analysis of Traffic Assignment Techniques with Actual Highway Use, National Cooperative Highway Research Program Report, no 58, 1968.

Hyman, G.M., The Calibration of Trip Distribution Models, Environment and Planning, Vol 1, pp. 105-112, 1969.

Irwin, N.A., Dodd, N. et von Cuke, H.G., Capacity Restraint in Assignment Programs, Bulletin Highway Research Bd. 297, pp. 109-127, 1961.

Jardine, N. et Sibson, R., Mathematical Taxonomy, John Wiley and Sons, Inc., 1971.

Jewell, W.S., Models for Traffic Assignment, Transportation Research, Vol. 1, pp. 31-46, 1967.

Kaufmann, A. et Desbazeille, G., La méthode du chemin critique, deuxième édition, Dunod, 1969.

Kleindorfer, George, B., Bounding Distributions for a Stochastic Acyclic Network, Operations Research, Vol. 19, no 7, 1971.

Lawton, W.H. et Sylvestre, E.A., Elimination of Linear Parameters in Non Linear Regression, Technometrics, Vol. 13, no 3, pp. 461-467, 1971.

Lefebvre, Jacques, Traitement des réseaux sur ordinateur, Publication no 72, Département d'informatique, Université de Montréal, 1971.

Leventhal, T.L., Newhauser, G.L. et Trotter, L.E., A Column Generation Algorithm for Optimal Traffic Assignment, Transportation Science, Vol. 7, no 2, 1973.

Lieberman, E., DYNET: a Dynamic Network Simulation of Urban Traffic Flow, "Proceedings of the Third Annual Simulation Symposium", 1970.

Lindsey, J.H. II, An Estimate of Expected Critical-Path in PERT Network, Operations Research, Vol. 20, no 4, pp. 800-812, 1971.

MacCrimmon, K.R. et Ryavec, C.A., An Analytical Study of the PERT Assumptions, Operations Research, Vol. 12, no 1, pp. 16-37, 1964.

MacLaughlin, W.A., Multi-path System Traffic Assignment, Report RB 108, Department of Highways, Ontario, Canada, 1966.

Malcolm, P.G., Roseboom, J.H., Clark, C.E. et Fazar, W., Application of a Technique for Research and Development, Operations Research, Vol. 7, pp. 646-669, 1959.

Martin, V.B., A Computer Program for Traffic Assignment Research, M.I.T. Civil Engineering Systems Laboratory, Research Report R. 64-41, 1964.

Martin, B.V. et Manheim, M.L., A Research Program for Comparison of Traffic Assignment Techniques, Highway Research Record 88, 1965.

Martin, J.J., Distribution of the Time Through a Directed, Acyclic Network, Operations Research, Vol. 13, no 1, pp. 46-66, 1965.

Moder, Joseph J. et Phillips, Cecil R., Project Management with CPM and PERT, deuxième édition, Van Nostrand Reinhold Company, 1970.

Mosher, W.W., A Capacity Restraint Algorithm for Assigning Flow to a Transport Network, Highway Research Record 88, 1963.

Moskowitz, K., California Method of Assigning diverted Traffic to proposed Freeways, Bulletin Highway Research Bd. 130, pp. 1-26, 1956.

Murchland, J.D., Road Network Traffic Distribution in Equilibrium, article présenté à la conférence "Mathematical Methods in the Economic Sciences", Mathematisches Forschungsinstitut, Oberwolfach, 1969.

Nguyen, Sang, Une approche unifiée des méthodes d'équilibre pour l'affectation du trafic, thèse de doctorat, Département d'informatique, Université de Montréal, décembre 1973.

Osofsky, S., A Multiple Regression Approach to Forecasting Urban Area Traffic Volumes, Traffic Quarterly, Vol. 13, no 3, pp. 423-, 1959.

Overgaard, K.R., Testing a Traffic Assignment Algorithm, Vehicular Traffic Science, Elsevier, New York, 1967.

Pospelov, G.S. et Barishpolets, V.A., A Stochastic PERT System. I, Engineering Cybernetics, no 6, 1966.

Potts, R.B. et Oliver, R.M., Flows in Transportation Networks, Academic Press, 1972.

Powell, M.J.D., An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives, The Computer Journal, Vol. 7, no 2, 1964.

Pritsker, Alan B. et Kiviat, Philip J., Simulation with GASP II, Prentice-Hall, Inc., 1969.

Rao, Radhakrishna C., Linear Statistical Inference and its Applications, deuxième édition, John Wiley and Sons, Inc., 1971.

Ringer, Larry J., Numerical Operators for Statistical PERT Critical Path Analysis, Management Science, Vol. 16, no 2, pp. B136-B143, 1969.

Robillard, Pierre, Probabilistic Assignment Method: an Analytic Study, Publication no 112, Département d'informatique, Université de Montréal, 1973.

Robillard, Pierre, Multipath Traffic Model with Dynamic Input Flows, Publication no 121, Département d'informatique, Université de Montréal, 1973.

Sakarovitch, M., The K Shortest chains in a Graph, Transportation Research, Vol. 2, pp. 1-11, 1968.

Schneider, M., A Direct Approach to Traffic Assignment, Highway Research Record 6, 1963.

Schneider, M., Probabilistic Maximization in Networks, papier présenté à l'"International Conference on Transportation Research", au Collège d'Europe, Bruges, juin 1973.

STOCH, Program Documentation, Urban Transportation Planning, U.S. Department of Transportation, Federal Highway Administration, 1972.

Stopher, Peter R., A Probability Model of Travel Mode Choice for the Work Journey, Highway Research Record 283, 1969.

Tillman, F.A., Pai, D.K. et Funk, M.L., An Iterative Approach to Traffic Assignment, Transportation Research, Vol. 2, pp. 63-72, 1968.

Tomlin, J.A., Minimum Cost Multicommodity Network Flows, Operations Research, Vol. 14, pp. 45-51, 1966.

Trueblood, D.L., Effect of Travel Time and Distance on Freeway uses, Bulletin Highway Research Bd. 61, pp. 18-37, 1952.

U.S. Department of Commerce, Bureau of Public Road, Traffic Assignment Manual, Washington, D.C., 1964.

U.S. Department of Transportation, STOCH: Program Documentation, Urban Transportation Planning, Federal Highway Administration, mars 1972.

U.S. Department of Transportation, Urban Transportation Planning General Information, Federal Highway Administration, mars 1972.

Van Slyke, R.M., Monte Carlo Methods and PERT Problem, Operations Research, Vol. 11, pp. 839-860, 1963.

Wardrop, J.G., Some Theoretical Aspects of Road Traffic Research, "Proceedings of the Inst. Civil Engineers, Part II", Vol. 1, pp. 325-378, 1952.

Wildermuth, Bruno R., The Use of a Multiple Routing Technique for One-Pass Capacity Restraint Assignment, Traffic Quarterly, pp. 247-265, 1972.

Wilkie, D.F. et Stefanek, R.G., Precise Determination of Equilibrium in Travel Forecasting Problems Using Numerical Optimization Techniques, Highway Research Record 369, 1971.

Wilson, A.G., Entropy in Urban and Regional Modelling, Pion Ltd., 1970.

Yagar, Samuel, Dynamic Traffic Assignment by Individual Path Minimization and Queuing, Transportation Research, Vol. 5, pp. 179-196, 1971.